

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA
CURSO DE ENGENHARIA ELETRÔNICA

CARLOS EDUARDO PORTO DE ALMEIDA SCHORK

**MEDIDOR ULTRASSÔNICO DE DISTÂNCIA PARA APLICAÇÃO EM
CONDUTORES DE DISTRIBUIÇÃO DE ENERGIA ELÉTRICA**

TRABALHO DE CONCLUSÃO DE CURSO

CAMPO MOURÃO

2019

CARLOS EDUARDO PORTO DE ALMEIDA SCHORK

MEDIDOR ULTRASSÔNICO DE DISTÂNCIA PARA APLICAÇÃO EM CONDUTORES DE DISTRIBUIÇÃO DE ENERGIA ELÉTRICA

Projeto de pesquisa apresentado à disciplina de Trabalho de Conclusão de Curso II (TCCII) do curso de Engenharia Eletrônica, do Departamento Acadêmico de Eletrônica (DAELN) do Campus Campo Mourão, da Universidade Tecnológica Federal do Paraná (UTFPR), como requisito parcial para a obtenção do título de Bacharel em Engenharia Eletrônica.

Orientador: Prof. Me. Lucas Ricken Garcia

CAMPO MOURÃO

2019



TERMO DE APROVAÇÃO DO TRABALHO DE CONCLUSÃO DE CURSO INTITULADO
MEDIDOR ULTRASSÔNICO DE DISTÂNCIA PARA APLICAÇÃO EM CONDUTORES DE
DISTRIBUIÇÃO DE ENERGIA ELÉTRICA

DO DISCENTE

CARLOS EDUARDO PORTO DE ALMEIDA SCHORK

Trabalho de Conclusão de Curso apresentado no dia 01 de Julho de 2019 ao Curso Superior de Engenharia Eletrônica da Universidade Tecnológica Federal do Paraná, Campus Campo Mourão. O discente foi arguido pela Comissão Examinadora composta pelos professores abaixo assinados. Após deliberação, a comissão considerou o trabalho aprovado, aprovado.

Prof. Dra. Jucélia Kuchla Vieira Gealh
UTFPR

Prof. Me. Osmar Tormena Junior
UTFPR

Prof. Me. Lucas Ricken Garcia
Orientador(a)
UTFPR

O Termo de Aprovação original, assinado, encontra-se na Coordenação do Curso.

DEDICATÓRIA

Dedico este trabalho de pesquisa à minha esposa Michele de Oliveira Schork, cujo o apoio incondicional possibilitou a conclusão dessa jornada. Também sou muito grato pela sua compreensão nos momentos ausentes. Te amo.

AGRADECIMENTOS

Gostaria de eternizar aqui meus agradecimentos às seguintes pessoas:

Minha família, pelo apoio, especialmente aos meus pais por terem me ensinado a importância da educação e a busca pelo conhecimento;

À minha sogra, Claudete, que durante a minha graduação auxiliou nos cuidados dos meus dois filhos;

Aos amigos que fiz nesta longa caminhada, e que me ajudaram com apoio e parceria, onde, por diversas vezes, através de grupos de estudos me livraram de algumas dificuldades.

A todos os professores que, ao longo dos anos, compartilharam seus conhecimentos comigo.

Ao meu Orientador, Prof. Me. Lucas Ricken Garcia, por toda ajuda, aconselhamento, orientação e motivação na empreitada desse trabalho de conclusão de curso.

RESUMO

Este trabalho apresenta detalhadamente a elaboração e desenvolvimento de um medidor ultrassônico de distância, portátil, para aplicação em condutores de distribuição de energia elétrica. Este instrumento foi idealizado devido à necessidade de se medir a altura mínima de condutores de energia elétrica de baixa, média e alta tensão em relação ao solo quando estes estão ligados, substituindo o uso de teodolitos e réguas de fibra de alta isolação, tendo em vista propiciar a segurança, a mobilidade e o conforto do profissional que realizará as medições. O aparelho realiza medições de distâncias através de sensores que emitem sinais ultrassônicos e detectam a reflexão destes sinais após a colisão com o objeto, ou seja, sem a necessidade de contato direto com o cabo energizado. Com a medição do tempo de emissão e detecção do sinal, refletido pelo objeto, é calculada a distância. Também foi implementada a medição da temperatura ambiente, pois os cabos das redes elétricas de distribuição sofrem dilatações consideráveis devido aos longos comprimentos dos vãos. Os principais componentes deste medidor são sensores ultrassônicos, sensor de temperatura, um *display* para interface com o operador e um microcontrolador para controlar e processar as leituras dos sensores e disponibilizá-las para o operador. Possui botões para navegação entre as funções de data e hora, medição das distâncias, medição da temperatura e consulta a registros que foram guardados na memória. O dispositivo obteve uma detecção máxima de aproximadamente 5 m para alvos com diâmetro de até 1 cm. Apesar de limitações quanto à resolução do sensor de ultrassom utilizado, o dispositivo desenvolvido ainda possui potencial para aplicação na detecção de cabos a menores distâncias.

Palavras-Chave: Sensor Ultrassônico, Redes de Distribuição, Serial RS-232, Altura de Cabos, Temperatura Ambiente.

ABSTRACT

This work presents the elaboration and development of a portable ultrasonic distance meter in detail for electric power distribution conductors' application. This instrument was designed due to the need to measure the minimum height of conductors of low, medium and high voltage electrical energy in relation to the ground, when they are switched on, replacing the use of theodolites and high insulation fiber measuring rod, given that to providing safety, the mobility and the comfort of the professional who will carry out the measurements. The device performs distance measurements through sensors that emit ultrasonic signals and detect the reflection of these signals after the collision with the object, that is, without the need for direct contact with the energized cable. By measuring the emission time and signal detection, reflected by the object, the distance is calculated. The measurement of the ambient temperature was also implemented, since the cables of the electricity distribution networks suffer considerable dilatation due to the long spans lengths. The main components of this meter are ultrasonic sensors, temperature sensor, a *display* for operator interface and a microcontroller to control and process the readings of the sensors and make them available to the operator. It has buttons for navigating between the date and time functions, measurement of distances, temperature measurement and records consultation that have been stored in memory. The device obtained a maximum detection of approximately 5 m for targets with a diameter of up to 1 cm. Regardless of resolution limitations of the ultrasound sensor used, the developed device still has potential for application in detecting cables at smaller distances.

Key-words: Ultrasonic Sensor, Distribution Networks, Serial RS-232, Cable Height, Ambient Temperature.

LISTA DE ILUSTRAÇÕES

| | |
|--|----|
| Figura 1 – Aparelho CHM300..... | 15 |
| Figura 2 – Aparelho AR600E..... | 16 |
| Figura 3 – Estrutura básica do sistema elétrico de potência..... | 18 |
| Figura 4 – Disposição dos cabos..... | 19 |
| Figura 5 – Distâncias dos cabos em relação ao solo..... | 20 |
| Figura 6 – Vara de Manobra Telescópica de Medição..... | 21 |
| Figura 7 – Exemplo de medição da altura de um objeto relacionado ao ângulo..... | 22 |
| Figura 8 – Teodolito analógico e digital..... | 23 |
| Figura 9 – Transmissão e recepção de ondas sonoras..... | 24 |
| Figura 10 – Formato do feixe ultrassônico..... | 25 |
| Figura 11 – Algumas aplicações de sensores ultrassônicos..... | 25 |
| Figura 12 – Medição da altura do cabo com o dispositivo ultrassônico..... | 27 |
| Figura 13 – Diagrama de blocos..... | 28 |
| Figura 14 – Mapa da área de alcance do sensor ultrassom ME007-ULA..... | 31 |
| Figura 15 – Sinal de dados do sensor utilizando o osciloscópio..... | 31 |
| Figura 16 – Identificação do frame de dados do sensor ME007-ULA V1 utilizando o software..... | 33 |
| Figura 17 – Identificação do frame de dados do sensor ME007-ULS V1 utilizando o software..... | 35 |
| Figura 18 – Circuito elevador de tensão..... | 36 |
| Figura 19 – Protótipo..... | 37 |
| Figura 20 – Navegação entre funções..... | 39 |
| Figura 21 – Fluxograma da implementação da função de leituras das distâncias..... | 40 |
| Figura 22 – Disposição do teclado..... | 40 |
| Figura 23 – Apresentação dos bytes em formato decimal no serial monitor da IDE do Arduino..... | 41 |
| Figura 24 – Resultado de medidas de cabos telefônicos em campo aberto..... | 44 |
| Figura 25 – Resultado de medidas entre dois muros..... | 44 |
| Figura 26 – Menu data e hora..... | 45 |
| Figura 27 – Menu distâncias..... | 45 |
| Figura 28 – Menu temperatura ambiente..... | 46 |
| Figura 29 – Menu consulta registro..... | 46 |

| | |
|--|----|
| Figura 30 – Placa de alimentação e botões..... | 47 |
| Figura 31 – Placa de alimentação e conexão de periféricos..... | 47 |
| Figura 32 – Placa dos botões..... | 48 |
| Figura 33 – Medidor finalizado | 49 |

LISTA DE QUADROS

| | |
|--|----|
| Quadro 1 – Detalhamento de cada caso da figura 3..... | 20 |
| Quadro 2 – Dados técnicos do sensor ME007-ULA V1..... | 30 |
| Quadro 3 – Formato do quadro do comando da comunicação serial..... | 32 |
| Quadro 4 – Formato do quadro de saída..... | 32 |
| Quadro 5 – Dados técnicos do sensor ME007-ULS V1..... | 34 |
| Quadro 6 – Resultados de distâncias e temperaturas..... | 42 |

LISTA DE TABELAS

| | |
|--|----|
| Tabela 1 – Níveis de tensão do Sistema de Distribuição Brasileiro..... | 18 |
| Tabela 2 – Tensões Nominais Padronizadas de Baixa Tensão..... | 18 |

LISTA DE SIGLAS

| | |
|--------------|--|
| A | ampere |
| ABNT | Associação Brasileira de Normas Técnicas |
| ANEEL | Agência Nacional de Energia Elétrica |
| AWG | American Wire Gauge (Escala Americana de Bitolas de Fios) |
| CA | Corrente Alternada |
| CC | Corrente Continua |
| GPS | Global Positioning System (Sistema de Posicionamento Global) |
| Hz | hertz |
| IDE | Integrated Development Environment (Ambiente de Desenvolvimento Integrado) |
| I/O | Input/Output (Entrada/Saída) |
| kHz | Kilo Hertz |
| kV | Kilo Volts |
| LED | Light Emitting Diode (Diodo Emissor de Luz) |
| m | metro |
| m/s | Velocidade no Sistema Internacional |
| NBR | Norma Brasileira |
| NTC | Normas Técnicas Copel |
| PRODIST | Procedimentos de Distribuição de Energia Elétrica no Sistema Elétrico Nacional |
| RTC | Real Time Clock (Relógio de Tempo Real) |
| s | Segundo |
| SEP | Sistema de Elétrico de Potencia |
| TTL | Transistor-Transistor Logic (Lógica Transistor-Transistor) |
| μ A | Micro Amper |
| V | Volt |
| $^{\circ}$ C | Graus Celsius |

SUMÁRIO

| | |
|---|-----------|
| 1 INTRODUÇÃO..... | 13 |
| 1.1 Objetivos..... | 14 |
| 1.1.1 Objetivo Geral..... | 14 |
| 1.1.2 Objetivos Específicos..... | 14 |
| 1.2 Justificativa..... | 14 |
| 2 FUNDAMENTAÇÃO TEÓRICA..... | 17 |
| 2.1 SEP..... | 17 |
| 2.2 Normas de Disposição de Cabos..... | 19 |
| 2.3 Equipamentos de medição atualmente mais utilizados..... | 20 |
| 2.3.1 Vara de Manobra Telescópica de Medição..... | 20 |
| 2.3.2 Teodolito..... | 22 |
| 2.4 Tecnologia Ultrassônica..... | 23 |
| 2.5 Temperatura Ambiente..... | 26 |
| 3 METODOLOGIA..... | 27 |
| 3.1 Concepção do medidor..... | 27 |
| 3.2 O microcontrolador..... | 28 |
| 3.3 Os sensores..... | 29 |
| 3.3.1 Sensores ultrassônicos..... | 29 |
| 3.3.2 Sensor de Temperatura..... | 36 |
| 3.4 Modulo RTC..... | 37 |
| 3.5 Display..... | 37 |
| 3.6 Protótipo..... | 37 |
| 4 RESULTADOS E DISCUSSÕES..... | 41 |
| 5 CONCLUSÃO..... | 50 |
| REFERÊNCIAS..... | 52 |
| ANEXOS..... | 55 |
| ANEXO A – Circuitos desenvolvidos e utilizados..... | 55 |
| ANEXO B – Imagens dos sensores e módulos utilizados..... | 56 |
| ANEXO C – E-mail de orçamento de dois modelos de sensores ultrassônicos solicitados à empresa SENSE..... | 58 |
| ANEXO D – Código de programa implementado na IDE do ARDUINO..... | 59 |

1 INTRODUÇÃO

O Sistema Elétrico de Potência, conhecido como SEP, se divide em três etapas: geração, transmissão e distribuição (ELETROBRAS, 2018c).

A geração é a etapa formada pelas unidades geradoras de energia elétrica, sejam elas hidrelétricas, termoelétricas, eólicas, etc. (ELETROBRAS, 2018a). A transmissão é a etapa responsável pelo transporte da energia elétrica gerada e que, por meio de torres e cabos, possibilita a entrega a longas distâncias da energia até as subestações nas cidades (FREITAS, s/d) (ELETROBRAS, 2018b). A terceira etapa, a distribuição, é onde a energia é direcionada em níveis mais baixos de tensão para a entrega às indústrias, comércios, hospitais, residências, etc. (ANEEL, 2018b).

Um sistema de distribuição de energia elétrica é constituído por várias redes que podem interligar e alimentar cidades e se ramificam pelas áreas rurais adentro, levando energia elétrica a todo lugar (ANEEL, 2018b).

As redes elétricas convencionalmente têm seus cabos de alumínio com diversas bitolas em AWG e, em sua grande maioria, são cabos nus, ou seja, são desencapados. Portanto, existem distâncias normatizadas que regulamentam as alturas mínimas em relação ao solo destas linhas devido a diversos fatores como: travessias de veículos de transportes, máquinas agrícolas, pedestres, e etc. (COPEL, 2018a).

Para garantir estas alturas, durante a construção de uma nova rede ou, eventualmente em redes existentes, torna-se necessário realizar a medição da distância do ponto mais baixo do cabo até o solo, ou seja, da flecha formada pelo vão entre duas torres ou dois postes (COPEL, 2018c). Também é tomada a temperatura ambiente, pois os cabos apresentam diferentes pesos para o tracionamento conforme a temperatura atual (COPEL, 2018b).

Esta medição normalmente é realizada por meio de instrumentos específicos como teodolitos e varas de medição de alta isolação elétrica e termômetros. A vara de medição é o instrumento mais utilizado. Estes instrumentos são muitas vezes difíceis e desajeitados para se transportar e utilizar devido ao peso e tamanho, tendo em vista maiores dificuldades em áreas rurais onde o local é composto de pastagens, lavouras ou até mesmo por uma mata (COPEL, 2016a) (COPEL, 2018b).

Diante disso, foi percebida a oportunidade do desenvolvimento de um instrumento eletrônico portátil, de baixo custo, que realize a medição da altura do cabo

em relação ao solo sem que se tenha contato com os cabos (no caso da vara) e que meça temperatura atual.

1.1 Objetivos

Nesta seção serão apresentados o objetivo geral e os objetivos específicos.

1.1.1 Objetivo Geral

Desenvolver um dispositivo para realizar a medição da altura de cabos de rede elétrica por meio de ultrassom e medir a temperatura ambiente.

1.1.2 Objetivos Específicos

- Obter protótipo funcional por meio da integração de microcontrolador e sensores;
- Possibilitar testes com a medição da altura de cabos de baixa e média tensão sem que haja contato e medir a temperatura utilizando-se o protótipo.
- Facilitar o processo de medição tradicional, mantendo a segurança do procedimento e precisão da medição.

1.2 Justificativa

A medição da altura dos cabos das redes de distribuição ou transmissão de energia elétrica é uma tarefa simples desde que estejam desligadas, pois assim, não há o risco de choque elétrico quando da utilização de réguas de medição ou trenas.

Muitas vezes, torna-se necessário a aferição da altura mínima de uma rede existente, em relação ao solo, para obedecer às distâncias mínimas para o tráfego de pessoas ou veículos, para a construção de outras redes que irão cruzar por baixo ou por cima dela, ou até mesmo a construção de uma via (COPEL, 2016a).

Difícilmente se desliga uma rede apenas para medir a sua altura. Diante disso, convencionalmente, utiliza-se de outras formas de medição um pouco mais complexas ou mais caras, como por exemplo, a utilização de teodolitos ou varas de medição de

fibra de vidro com luvas de borracha (a vara e as luvas devem possuir alta isolamento elétrica devido ao contato direto com os cabos).

Devido ao efeito de dilatação e contração dos cabos de redes elétricas, as diferentes temperaturas influenciam no tracionamento dos cabos (COPEL, 2018b).

O desenvolvimento de um dispositivo que meça a distância de um cabo elétrico até o solo, sem que haja contato com o objeto, utilizando a tecnologia de ultrassom para se obter um resultado instantaneamente, e ao mesmo tempo meça a temperatura ambiente, facilitará e tornará a medição mais ágil e mais segura.

Em pesquisas realizadas para verificar a existência de um instrumento portátil similar, foram encontrados apenas dois aparelhos de fabricantes distintos e de fora do país, sendo necessária a importação deles. Um é o CHM300, da marca Suparule, com o preço de 472,00 Libras esterlinas (por volta de R\$ 2.360,00) (ZENITH SURVEY, 2018); já o segundo, é o AR600E, da marca Smartsensor, com o preço de 117,00 dólares (por volta de R\$ 500,00) (ALIEXPRESS, 2018a). Os valores são referentes apenas ao produto e não estão computados os valores referentes ao frete e taxas de importação. A seguir, as Figuras 1 e 2 apresentam os dois aparelhos.

Figura 1 – Aparelho CHM300



Fonte: ZENITH SURVEY (2018).

Figura 2 – Aparelho AR600E



Fonte: ALIEXPRESS (2018a).

2 FUNDAMENTAÇÃO TEÓRICA

Nesta seção são apresentadas as informações sobre o sistema elétrico de potência, distâncias dos cabos elétricos, equipamentos utilizados para medição de alturas dos cabos, tecnologia ultrassônica e da medição da temperatura.

2.1 SEP

A eletricidade é a principal fonte de luz, calor e força-motriz utilizada no mundo contemporâneo. Indústrias, comércios, universidades, centros de pesquisas e uma infinidade de outros lugares se utilizam dela para funcionar. Obtida a partir de outros tipos de energia, a eletricidade segue um caminho até os consumidores em todo mundo por meio de sistemas complexos, que se compõe de três etapas: geração, transmissão e distribuição (ELETROBRAS, 2018c). O conjunto destas etapas formam o Sistema Elétrico de Potência, também chamado SEP.

A primeira etapa é responsável por transformar outros tipos de energia em energia elétrica. No Brasil, a geração mais comum é a hidrelétrica, que utilizam a energia potencial das águas represadas para girar suas turbinas, na sequência as termoelétricas, que necessitam queimar combustíveis para aquecer caldeiras que, com a força do vapor movimentam as turbinas. Também pode-se citar outras gerações ainda menos utilizadas: as eólicas e solares (ELETROBRAS, 2018a).

A etapa responsável pelo transporte da energia elétrica das usinas geradoras às grandes áreas de consumo, e que tem suas tensões elevadas por subestações chamadas elevadoras, para que se possa transportar a eletricidade a longas distâncias com o mínimo de perdas possível, chama-se transmissão. Após a elevação da tensão, basicamente o sistema se constitui por torres e cabos formando diversas linhas chamadas de linhas de transmissão, que no Brasil, possuem níveis de tensão entre 230 kV e 750 kV CA e 600 kV e 800 kV CC (FREITAS, s/d) (ELETROBRAS, 2018b).

Na etapa de distribuição, os níveis de tensões são reduzidos por subestações abaixadoras a níveis compatíveis de entrega aos consumidores urbanos e rurais. O sistema é composto por torres e cabos para alta tensão, e postes, cabos e transformadores para média e baixa tensão. Conforme estabelece o órgão regulador, a

Agência Nacional de Energia Elétrica (ANEEL), os níveis de tensões padronizados estão descritos nas Tabelas 1 e 2 (ANEEL, 2018b).

Tabela 1 – Níveis de tensão do Sistema de Distribuição Brasileiro

| CLASSIFICAÇÃO | TENSÃO |
|-------------------|---|
| ALTA TENSÃO (AT) | $\geq 69 \text{ kV}$ e $< 230 \text{ kV}$ |
| MEDIA TENSÃO (MT) | $> 1 \text{ kV}$ e $< 69 \text{ kV}$ |
| BAIXA TENSÃO (BT) | $\leq 1 \text{ kV}$ |

Fonte: ANEEL (2018b).

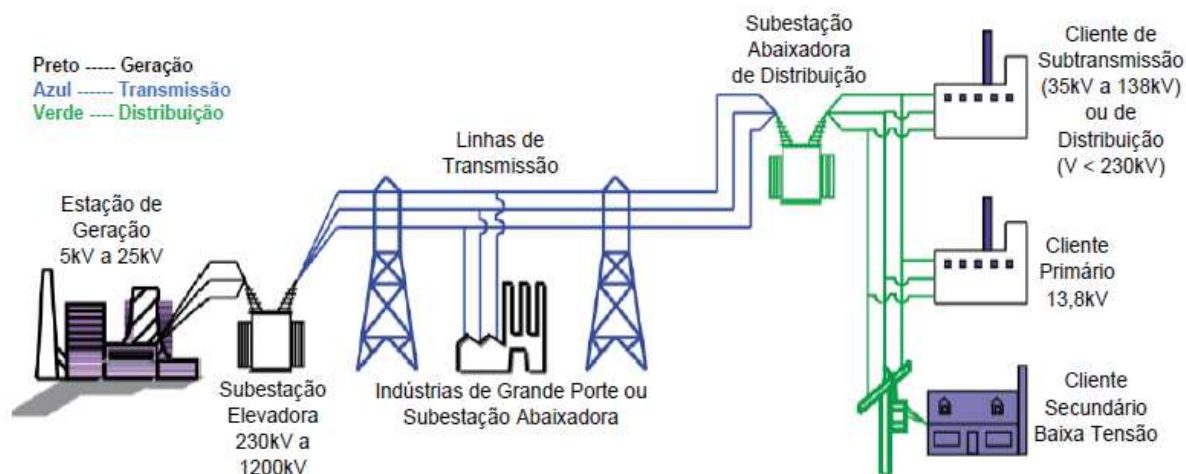
Tabela 2 – Tensões Nominais Padronizadas de Baixa Tensão

| Sistema | Tensão Nominal (V) |
|------------|--------------------|
| Trifásico | 220/127 |
| | 380/220 |
| Monofásico | 254/127 |
| | 440/220 |

Fonte: ANEEL (2018b).

Abaixo, a figura 3 detalha cada etapa do SEP.

Figura 3 – Estrutura básica do sistema elétrico de potência

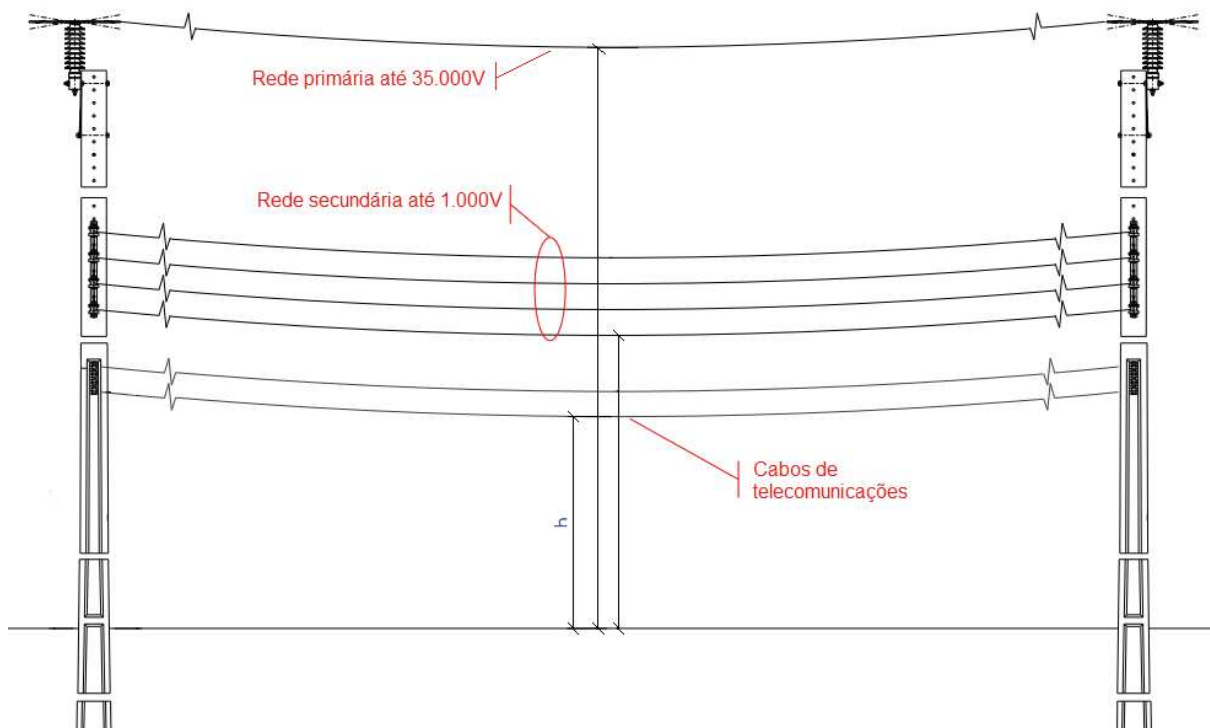


Fonte: FREITAS (s/d).

2.2 Normas de Disposição de Cabos

O dispositivo proposto neste trabalho deverá realizar a medição da distância de um cabo de rede elétrica ou de telecomunicações em relação ao solo, conforme pode ser visto na Figura 4 abaixo:

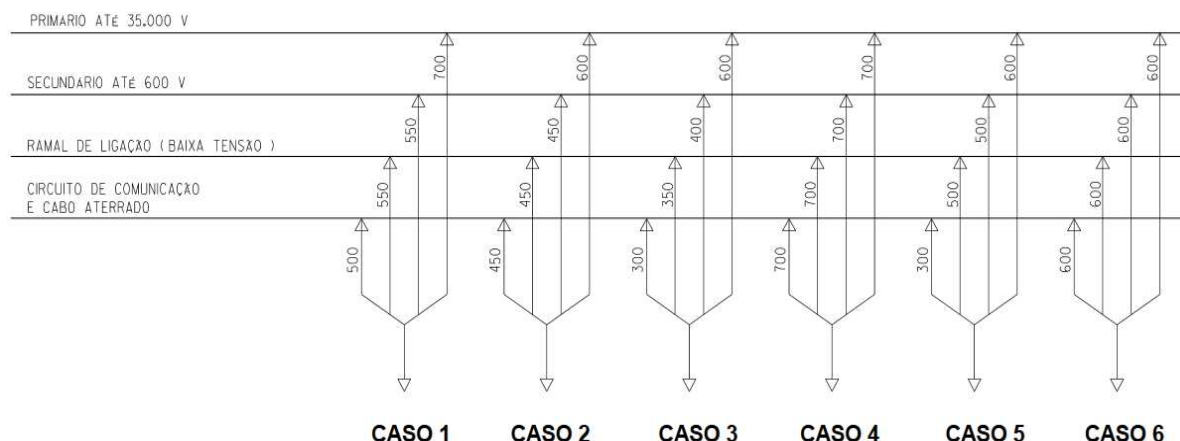
Figura 4 – Disposição dos cabos



Fonte: COPEL (2018c).

Estas distâncias são regulamentadas por normas, e como exemplo pode-se verificar na Figura 5, onde a concessionária Copel – Companhia Paranaense de Energia, estabelece suas distâncias mínimas, que são baseadas nas normas brasileiras, Norma ABNT NBR 15688:2012 (COPEL, 2018c).

Na Figura 5, as medidas estão em centímetros e abaixo, no Quadro 1 segue o detalhamento de cada caso.

Figura 5 – Distâncias dos cabos em relação ao solo

Fonte: COPEL (2016a).

Quadro 1 – Detalhamento de cada caso da Figura 5

| CASO: | DESCRIÇÃO |
|--------------|--|
| Caso 1 | Ruas e Avenidas |
| Caso 2 | Entradas de prédios e demais locais de uso restrito a veículos. |
| Caso 3 | Ruas e vias exclusivas a pedestres. |
| Caso 4 | Rodovias. |
| Caso 5 | Área rural "A" (local acessível exclusivamente à pedestres). |
| Caso 6 | Área rural "B" (Local acessível a trânsito de veículos, máquinas agrícola e travessias sobre estradas particulares). |

Fonte: COPEL (2016a).

2.3 Equipamentos de medição atualmente mais utilizados

Atualmente, para a realização de medições das alturas dos cabos de linhas de transmissão e redes de distribuição já existentes e energizadas, utilizam-se equipamentos como teodolitos e varas de fibra de vidro de alta isolamento. Esta última, tem seu uso acompanhado de luvas de borracha, também de alta isolamento, para maior proteção do operador, tendo em vista que haverá o contato da ponta da vara com o cabo ao medir sua altura (RITZMG, 2019) (ESCOLA, 2018).

2.3.1 Vara de Manobra Telescópica de Medição

A vara de manobra telescópica, também chamada de VTT, tem por objetivo garantir a distância de segurança e o isolamento necessário nas medições em instalações elétricas, sendo, portanto, uma ferramenta e ao mesmo tempo um equipamento de segurança dos mais usados nesse setor (RITZMG, 2019).

A vara é fabricada em tubo de fibra de vidro impregnada com resina epóxi. Possui marcações numéricas gravadas de 10 em 10 cm e somente marcas intermediárias a cada centímetro, assim, para realizar a leitura, basta apoiar a vara no solo em posição vertical e à medida que os elementos são deslocados para cima, permite ao operador verificar a medida indicada (RITZMG, 2019).

A utilização da VTT dispensa o uso de escadas ou plataformas, possibilitando a execução da tarefa diretamente do solo (RITZMG, 2019). Este equipamento pode ser comprado em lojas de produtos e equipamentos do ramo do setor elétrico e com valores a partir de R\$ 1200,00. A Figura 6 apresenta o equipamento.

Figura 6 – Vara de Manobra Telescópica de Medição

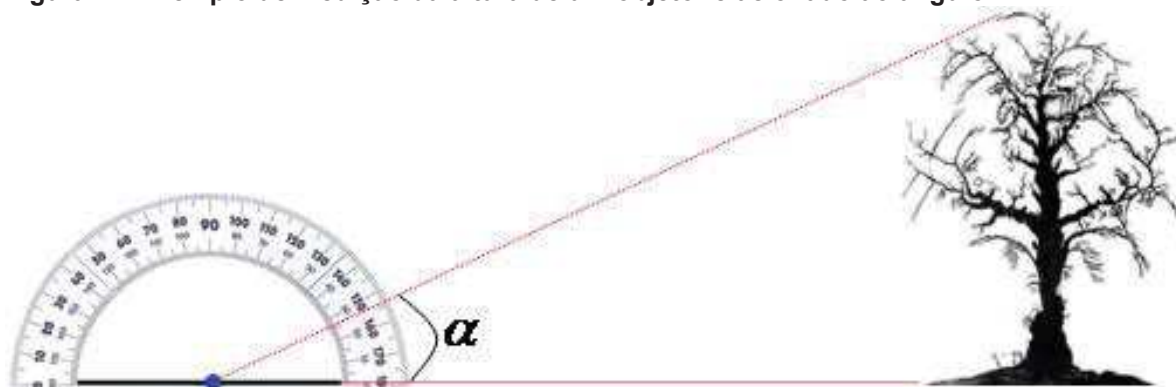


Fonte: D'ASSIS (2018).

2.3.2 Teodolito

O teodolito é um instrumento de precisão óptico que mensura ângulos verticais e horizontais, nos quais, aplicam-se as regras da trigonometria para obter-se distâncias, áreas e alturas. A Figura 7 apresenta uma medição desse tipo.

Figura 7 – Exemplo de medição da altura de um objeto relacionado ao ângulo



Fonte: ESCOLA (2018).

Este instrumento é bastante utilizado por engenheiros, agrimensores, topógrafos e antigos navegadores para medir distâncias inacessíveis. É aplicado em diversos setores como na navegação, na construção civil, na agricultura e outras diversas áreas (ESCOLA, 2018).

Existem diversos modelos, desde os mais simples, manuais, até os mais sofisticados que são equipados com GPS e computadores embutidos em sua estrutura (ESCOLA, 2018).

Os teodolitos eletrônicos podem ser encontrados com valores a partir de R\$ 4.000,00. Teodolitos manuais/analógicos já são mais difíceis de encontrar novos. A Figura 8 exemplifica alguns modelos de teodolitos (PHILODENDROS, 2018).

Figura 8 – Teodolito analógico e digital



Fonte: PHILODENDROS (2018).

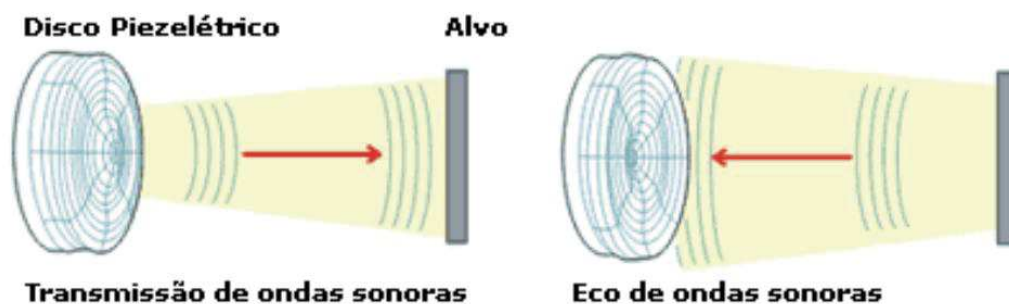
2.4 Tecnologia Ultrassônica

A tecnologia ultrassônica permite realizar a medição de distâncias de objetos com bastante exatidão. Transdutores ultrassônicos são construídos com cristais piezoelétricos, pois estes, têm a capacidade de ressonar em altas frequências e são capazes de converter energia elétrica em energia acústica e vice-versa (COSTA, 2016).

Os sensores ultrassônicos funcionam com a emissão de uma onda sonora de alta frequência cíclica, geralmente frequências acima de 40 kHz, que ao se chocar com determinado objeto a reflete de volta ao sensor, para que este, ao perceber o eco (retorno da onda sonora emitida), gere um sinal elétrico em sua saída (THOMAZINI, 2012).

Este sinal elétrico, na saída do sensor, é utilizado para se calcular a distância do objeto utilizando o cálculo do tempo em que a onda foi emitida e recebida novamente (THOMAZINI, 2012). A Figura 9 exemplifica o processo.

Figura 9 – Transmissão e recepção de ondas sonoras



Fonte: AUTOMATIZE (2018).

A distância pode ser obtida através do seguinte cálculo, dado pela equação (1):

$$d = \frac{t.v}{2} \quad (1)$$

Onde:

d = distância;

t = tempo de emissão e recepção da onda;

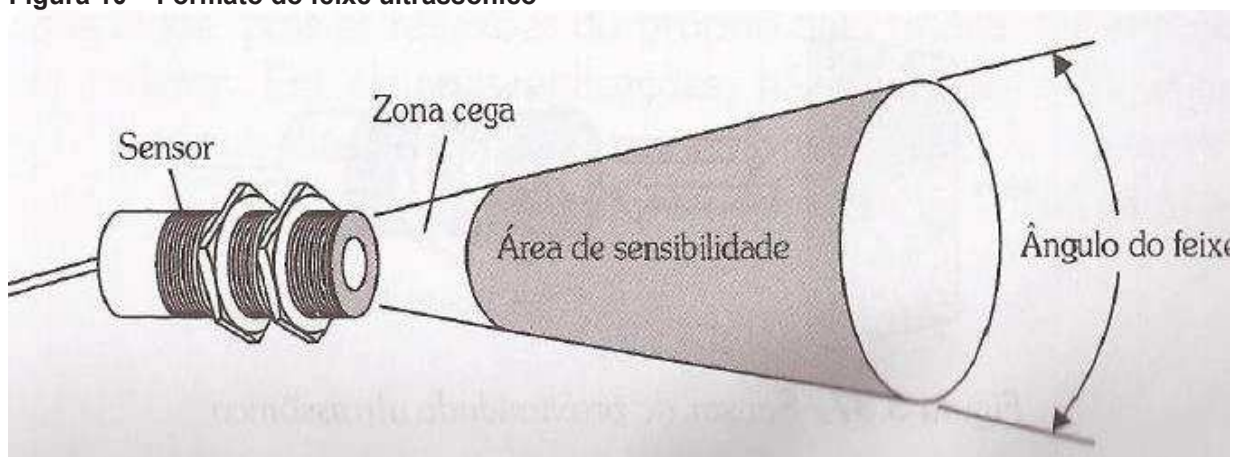
v = velocidade do som.

Considerando que a velocidade do som é igual a 340 m/s e o tempo em segundos, a distância será obtida em metros. A divisão por dois, na fórmula, se deve ao fato de que a onda percorre o caminho duas vezes, uma vez para ida e outra para volta (THOMAZINI, 2012).

Todo sensor ultrassônico possui uma zona morta, ou seja, uma distância mínima para que o sensor possa prover um atraso no tempo de forma que os “ecos” possam ser interpretados. Algumas variáveis podem interferir na operação de um sensor ultrassônico, como o ângulo da superfície do objeto, rugosidade da superfície, mudanças na temperatura ou umidade do ar. Os objetos podem ter qualquer forma refletiva, e até podem ser redondos (THOMAZINI, 2012).

As ondas sonoras são transmitidas em forma de cone, como pode ser visto na Figura 10, e a reflexão também segue o mesmo padrão (COSTA, 2016).

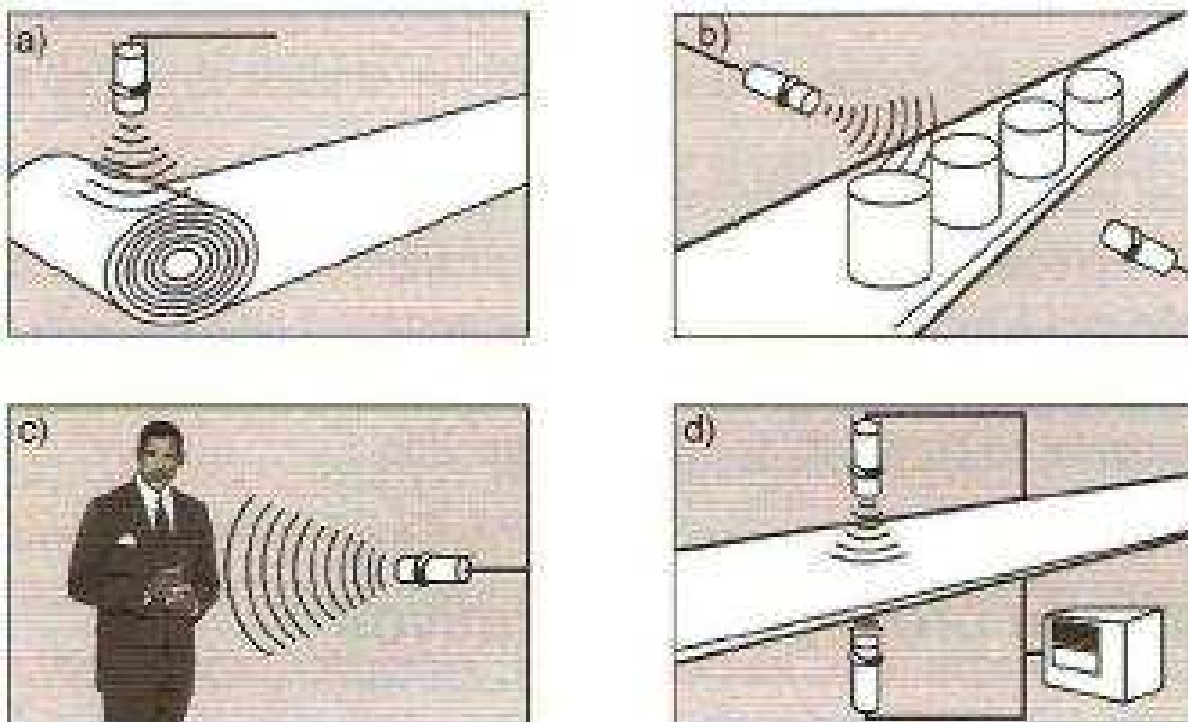
Figura 10 – Formato do feixe ultrassônico



Fonte: COSTA (2016).

Os sensores ultrassônicos têm diversas aplicações, a Figura 11 ilustra algumas delas como, medida de diâmetro de rolos (a), detecção e contagem de objetos (b), presença de pessoas (c), medição de densidade (d) etc.

Figura 11 – Algumas aplicações de sensores ultrassônicos



Fonte: THOMAZINI (2012).

2.5 Temperatura Ambiente

A medição da temperatura ambiente se faz necessária para o auxílio na elaboração de projetos e na execução de obras de redes elétricas (COPEL, 2018b).

Os cabos de linhas e redes elétricas são constituídos, na maioria das vezes, de alumínio, material este, que possui um coeficiente de dilatação considerável, em termos de construção de redes de energia. Esta dilatação influencia no tracionamento do próprio cabo na estrutura de ancoragem e em relação a sua flecha (barriga) no vão (COPEL, 2018b).

Para construção de uma rede, por baixo ou por cima de uma já existente, deve-se considerar a temperatura para que, em épocas de baixas ou altas temperaturas, não ocorra o risco das redes se tocarem devido ao efeito de dilatação (COPEL, 2018b).

Em dias extremamente frios os cabos tendem a se contraírem, podendo ocasionar o próprio rompimento e também danificar as estruturas de ancoragens, que são constituídas por isoladores de porcelana e postes (COPEL, 2016b). Já em dias mais quentes, com a dilatação dos cabos a flecha tende a aumentar, ou seja, os cabos baixam no meio do vão.

Diante disso, para a elaboração de projetos e construção de redes elétricas são utilizados valores calculados de trações para as montagens dos cabos de acordo com a temperatura, bitola e o comprimento do vão (COPEL, 2015; COPEL, 2018b).

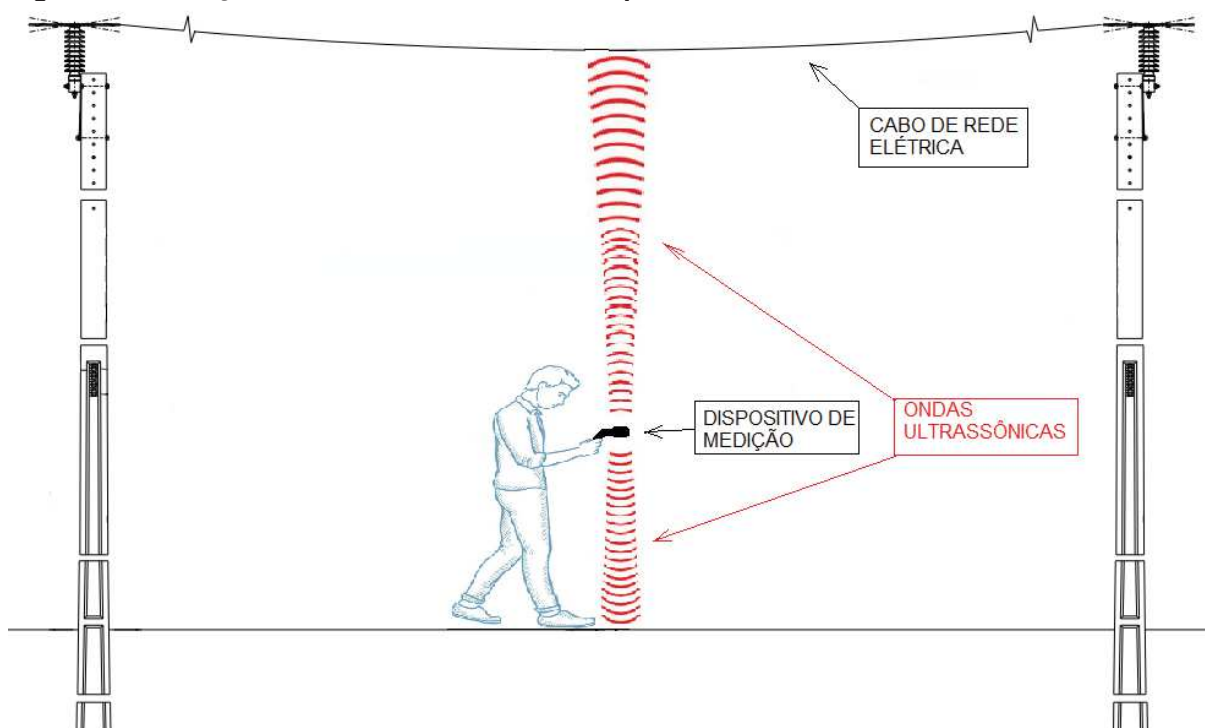
3 METODOLOGIA

São descritos nesta seção os materiais utilizados e procedimentos realizados neste projeto.

3.1 Concepção do medidor

Com a idéia de se desenvolver um aparelho portátil, chegou-se a um modelo que utiliza dois sensores ultrassônicos: um detecta a distância do aparelho que está na mão do operador até o chão e outro sensor que detecta a distância do aparelho até o cabo que se pretende verificar a altura, obtendo através da soma das duas distâncias o resultado final que é a distância do solo até o cabo, conforme ilustrado na Figura 12.

Figura 12 – Medição da altura do cabo com o dispositivo ultrassônico



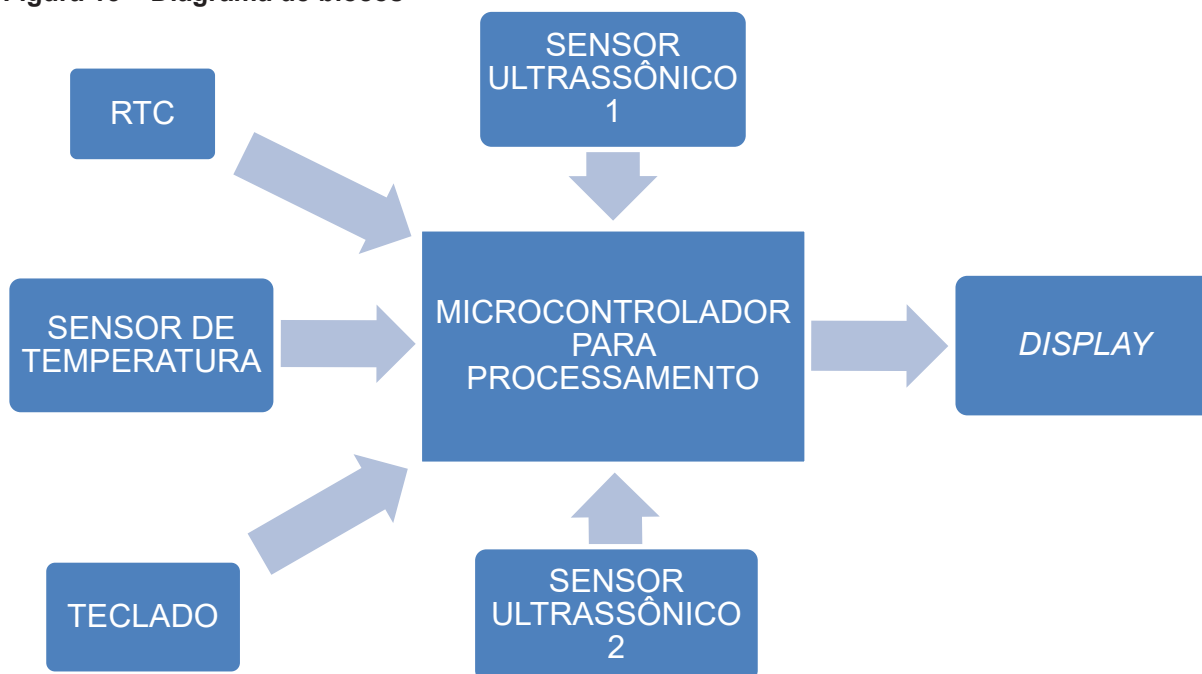
Fonte: Adaptado de COPEL (2018c).

O dispositivo foi implementado também com um sensor de temperatura ambiente.

Um microcontrolador será responsável por processar os sinais obtidos dos sensores ultrassônicos e do sensor de temperatura e disponibilizar as informações de distâncias e temperatura em um *display* para o operador no momento da medição. A

Figura 13 exemplifica resumidamente em diagrama de blocos os principais componentes e suas funções como entradas e saídas.

Figura 13 – Diagrama de blocos



Fonte: Autoria própria (2018).

3.2 O microcontrolador

Inicialmente foi utilizado no projeto uma placa Arduino UNO, que possui um microcontrolador Atmega328P, com relógio de 16 MHz, 14 entradas/saídas digitais e mais 6 entradas analógicas. Devido a implementação de 2 sensores ultrassônicos, 1 sensor de temperatura, 1 *display* de cristal líquido, 1 módulo *Real Time Clock* (RTC), 5 botões e um *buzzer*, foi necessário substituir o Arduino UNO por um Arduino Mega 2560. Este é um modelo de placa que possui o microcontrolador ATmega2560, com relógio de 16 MHz, 54 entradas/saídas digitais e 16 entradas analógicas.

Para a programação foi utilizada a IDE do Arduino, que utiliza uma linguagem baseada em C/C++. O código de programa foi concluído com um total de 746 linhas.

3.3 Os sensores

Nesta seção serão apresentadas as informações referentes aos sensores de distância e de temperatura utilizados.

3.3.1 Sensores ultrassônicos

Os sensores ultrassônicos, em sua grande maioria, são de curtas distâncias, com alcance entre dois a quatro metros. Conforme a sua precisão, resolução e alcance o custo aumenta consideravelmente.

Primeiramente, para este projeto, foi utilizado um sensor ultrassônico de longo alcance, o ME007-ULA V1. Este sensor de fabricação chinesa foi escolhido por apresentar as suas características conforme Quadro 2 e seu baixo custo. Os dados foram fornecidos pelo vendedor do sensor, mas não foi fornecido o *datasheet* ou manual pelo vendedor e nem pelo fabricante.

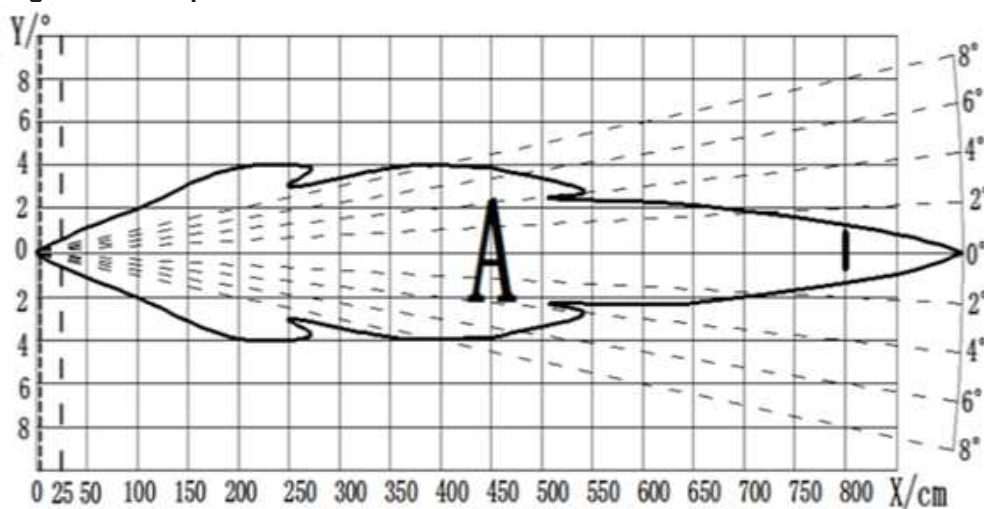
Quadro 2 – Dados técnicos do sensor ME007-ULA V1

| | RS-232 | Saída de tensão analógica |
|------------------------------|---|----------------------------|
| Tensão de alimentação | 5 V-12 V, <i>ripple</i> menor que 100 mV | |
| Corrente média | Tensão de alimentação 5 V/ ciclo de trabalho 500 ms, Corrente média 7m A | |
| Frequência central | 40 kHz | |
| Zona cega | 250 mm | |
| Detecção do ponto de partida | Precisa subtrair 40 mm (a distância da superfície do transdutor ultrassônico ao plano da boca da corneta) | |
| Max. distância de detecção | Mais que 8000 mm | <i>Default</i> até 3300 mm |
| Ângulo de emissão | 8° | |
| Ciclo de trabalho | <i>Default</i> 500 ms | |
| Precisão de teste | ± 1 cm | |
| Resolução de alcance | 1 mm | |
| Compensação de temperatura | Com compensação de temperatura | |
| Saída de temperatura | Tem | Não tem |
| Temperatura de operação | Min. -25 °C e max. 70 °C | |
| Temperatura de armazenamento | Min. -40 °C e max. 85 °C | |

Fonte: Adaptado de DOCPLAYER (2019).

A Figura 14 apresenta um mapa do alcance do sensor, bem como o ângulo de abertura.

Figura 14 – Mapa da área de alcance do sensor ultrassom ME007-ULA



Fonte: ALIEXPRESS (2018b).

Foram realizadas análises com um osciloscópio, para descobrir os sinais de saída, que neste sensor, pode-se alternar por uma saída digital serial, no padrão RS-232, ou uma saída analógica, em níveis de tensão de 0 a 5 V.

Através da saída digital o sensor consegue um alcance de 8000 mm, já na saída analógica, somente até 3300 mm. Optou-se por utilizar a saída digital, devido ao seu maior alcance.

Este sensor, também apresenta a medição de temperatura na saída digital.

Ao verificar no osciloscópio, a sua saída serial RS-232 apresentou um formato de dados conforme a Figura 15:

Figura 15 – Sinal de dados do sensor utilizando o osciloscópio



Fonte: Autoria própria (2019).

No osciloscópio foi possível identificar alguns bits e obter a velocidade de 9600 bits/s. Foi necessário realizar a conversão do sinal do padrão RS-232 para nível TTL 5V, utilizando o conversor MAX3232 para possibilitar a comunicação com o microcontrolador.

Ao realizar mais pesquisas sobre informações do sensor, foi encontrado apenas um manual com informações referente a outro modelo de sensor com características parecidas, o ME007-ULS V1, o qual foi adquirido posteriormente. O manual apresentou as seguintes informações descritas nos Quadros 3 e 4:

Quadro 3 – Formato do quadro do comando da comunicação serial

| | Tipo de interface | Bit de início | Bits de dados | Bit de parada | Paridade | Baud rate |
|------|--------------------|---------------|---------------|---------------|----------|-----------|
| UART | <i>Full duplex</i> | 1 | 8 | 1 | Nenhum | 9600bps |

Fonte: DOCPLAYER (2019).

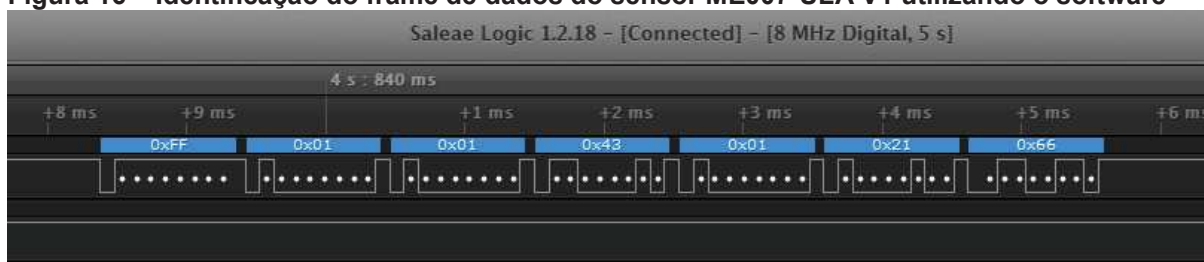
Quadro 4 – Formato do quadro de saída

| Dados do quadro | Instruções | Byte |
|--------------------------------------|---|--------|
| Identificação do cabeçalho do quadro | Fixado em 0xFF | 1 byte |
| Data_H | Byte mais significativo da distancia | 1 byte |
| Data_L | Byte menos significativo da distancia | 1 byte |
| Temp_H | Byte mais significativo da temperatura | 1 byte |
| Temp_L | Byte menos significativo da temperatura | 1 byte |
| Soma | A soma de verificação da comunicação | 1 byte |

Fonte: DOCPLAYER (2019).

Para auxiliar na identificação dos formatos dos quadros de dados com seus referidos bytes foi utilizado um analisador lógico para aquisição do sinal e o software Saleae Logic 1.2.18 que apresenta os dados do sinal, onde foi possível identificar corretamente os bytes e suas informações. A Figura 16 mostra um quadro de dados identificando sete bytes com a utilização do analisador lógico e o software representando o sinal.

Figura 16 – Identificação do frame de dados do sensor ME007-ULA V1 utilizando o software



Fonte: Autoria própria (2019).

O manual informa como se obtém a distância e a temperatura a partir da leitura dos bytes do quadro de dados, conforme demonstrado na Equação 2 a seguir.

$$\text{Valor da distância} = (\text{Data_H}) \times 256 + (\text{Data_L}) \quad (2)$$

Converte-se o resultado, de binário (ou hexadecimal) para decimal para se obter o valor da distância em mm. Para se obter o valor da temperatura realiza-se um cálculo parecido ao anterior, conforme Equação 3.

$$\text{Valor da temperatura} = (\text{Temp_H}) \times 256 + (\text{Temp_L}) \quad (3)$$

Converte-se o resultado em decimal e divide por dez, para se obter o valor da temperatura em graus Celsius.

Foi percebido que o ME007-ULA V1 possuía um byte a mais em relação ao outro modelo do manual, o segundo byte logo após o byte 0xFF, apresentado na Figura 16, o qual foi verificado que se tratava de um byte de endereço, pois nunca se alterava e sempre estava apresentando o valor 0x01, independente da distância e da temperatura.

Ao realizar testes de medidas de distâncias e de temperaturas verificou-se que o sensor ME007-ULA V1 disponibilizava em sua saída as informações em ciclos de 500 ms. Foi substituído o sensor ME007-ULA V1 pelo ME007ULS V1, que tem três tipos de sinais de saída. O Quadro 5 apresenta os dados deste sensor.

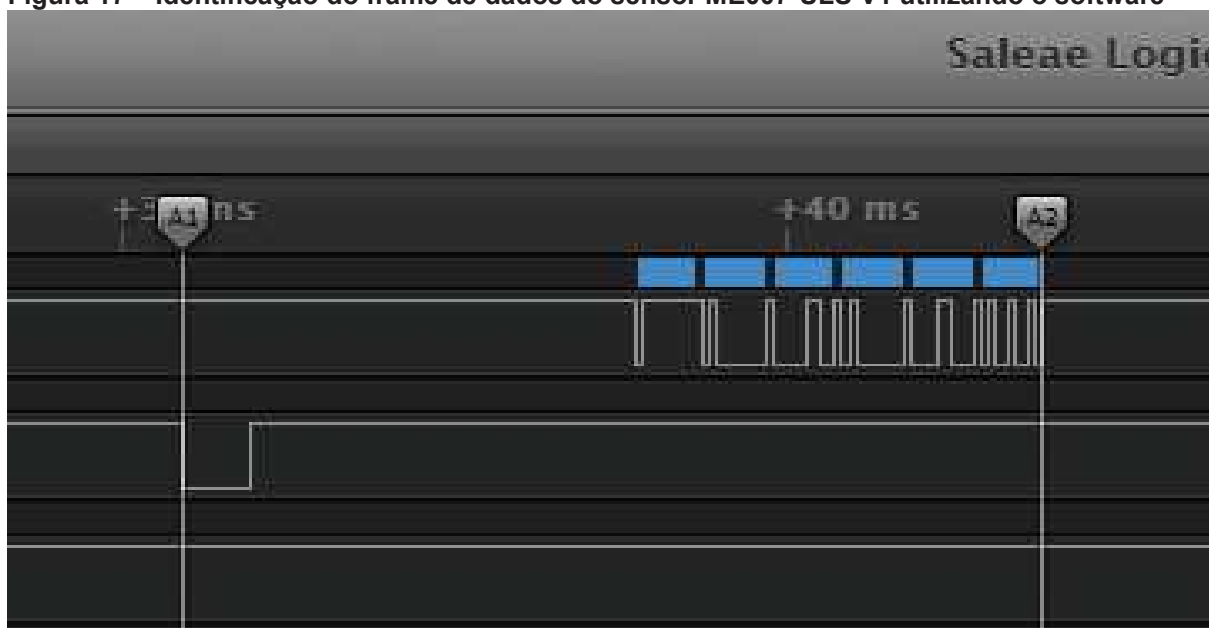
Quadro 5 – Dados técnicos do sensor ME007-ULS V1

| | Saída PWM | Saída serial | Saída digital |
|------------------------------|---|--------------|----------------------|
| Tensão de alimentação | 3.3 V-12 V | | |
| Corrente de <i>standby</i> | Menos que 200 μ A | | Mais que 400 μ A |
| Corrente média | 2 mA (baseado na tensão de alimentação 5 V ciclo de 250 ms) | | |
| Frequência central | 40 kHz | | |
| Zona cega | 250 mm | | |
| Max. distância de detecção | Mais que 8000 mm | | |
| Ângulo de emissão | 8° | | |
| Ciclo de trabalho | Mais que 60 ms, controlado pelo sinal de gatilho | | 250 ms |
| Sinal de gatilho | Borda de queda, nível baixo mantém 0.1-10 ms | | GND conectado |
| Precisão de teste | ± 1 cm | | |
| Resolução de alcance | 1 mm | | |
| Compensação de temperatura | Sem compensação | compensação | |
| Saída de temperatura | Não tem | Tem | Não tem |
| Temperatura de operação | Min. -25 °C e max. 70 °C | | |
| Temperatura de armazenamento | Min. -40 °C e max. 85 °C | | |

Fonte: DOCPLAYER (2018).

Ao substituir o sensor foi necessário ajustar o código do programa para ler somente seis bytes e também adicionado uma rotina para gerar um gatilho (*trigger*) de 1 ms, pois este sensor depende deste gatilho para apresentar em sua saída um quadro com as informações da distância e temperatura. A seguir, a Figura 17 apresenta a coleta do sinal contendo o quadro de dados do sensor ME007-ULS V1 após receber o gatilho.

Figura 17 – Identificação do frame de dados do sensor ME007-ULS V1 utilizando o software



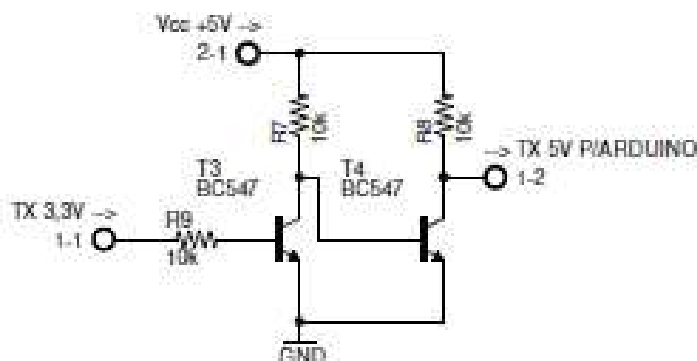
Fonte: Autoria própria (2019).

A substituição do sensor por este segundo modelo foi motivada pela velocidade do ciclo de leitura, que depende do gatilho, o qual foi implementado para gerar ciclos de no mínimo 61 ms, tempo esse, que conforme os dados informados anteriormente no Quadro 4, deve ser maior que 60 ms.

Com a geração de ciclos mais rápidos foi possível realizar varreduras de leituras para detectar as menores distâncias durante o posicionamento do aparelho em baixo de um cabo, o que não foi possível implementar com o sensor ME007-ULA V1 devido ao seu ciclo *default* de 500 ms.

Na substituição de um sensor por outro, foi verificado que o novo sensor trabalha com um nível de tensão do sinal de dados igual a 3,3 V. O Arduino utiliza um microcontrolador que trabalha com níveis TTL de 5 V em suas portas I/O. Para resolver o problema, foi implementado um circuito para elevar a tensão do sinal de 3,3 V para 5 V, conforme apresentado na Figura 18.

Figura 18 – Circuito elevador de tensão



Fonte: Autoria própria (2019).

Os sensores ME007-ULA V1 e ME007-ULS V1, por serem de longo alcance e possuírem resolução de 1 mm, foram implementados no projeto para obterem as distâncias para cima, ou seja, do aparelho na mão do operador até os cabos. Para realizar a medição do aparelho na mão do operador até o chão, foi utilizado o sensor HC-SR04. Este sensor, também ultrassônico, é bastante comum e de custo baixo. Tem uma faixa de detecção de 20 mm a 4000 mm e resolução de 3 mm.

Devido o HC-SR04 ser bastante utilizado em diversos tipos de projetos, possui vasto conteúdo com informações e exemplos de implementações com sua utilização, ao contrário dos sensores ME007-ULA V1 e ME007-ULS V1.

3.3.2 Sensor de Temperatura

Para a medição da temperatura foi utilizado o sensor LM35, que é um sensor de precisão com uma saída analógica linear e com tensão de alimentação de 4 a 20 V. Possui uma precisão de $\pm 0,5$ °C e tem uma tensão de saída de 10 mV para cada grau, ou seja, se a saída for 220 mV isto equivale à 22 °C.

Os dois sensores ultrassônicos, o ME007-ULA V1 e o ME007-ULS V1, disponibilizam a medição da temperatura, mas devido à baixa precisão não foi utilizada essa função.

Optou-se por implementar este sensor no projeto devido este tipo de sensor ter uma boa precisão, implementação simples e um custo baixo.

3.4 Módulo RTC

Neste projeto foi implementado um módulo RTC que utiliza o chip DS1302, que foi conectado ao Arduino para fornecer a data e a hora para o aparelho. Foi equipado com uma bateria independente para que não perca a contagem da data e da hora.

3.5 Display

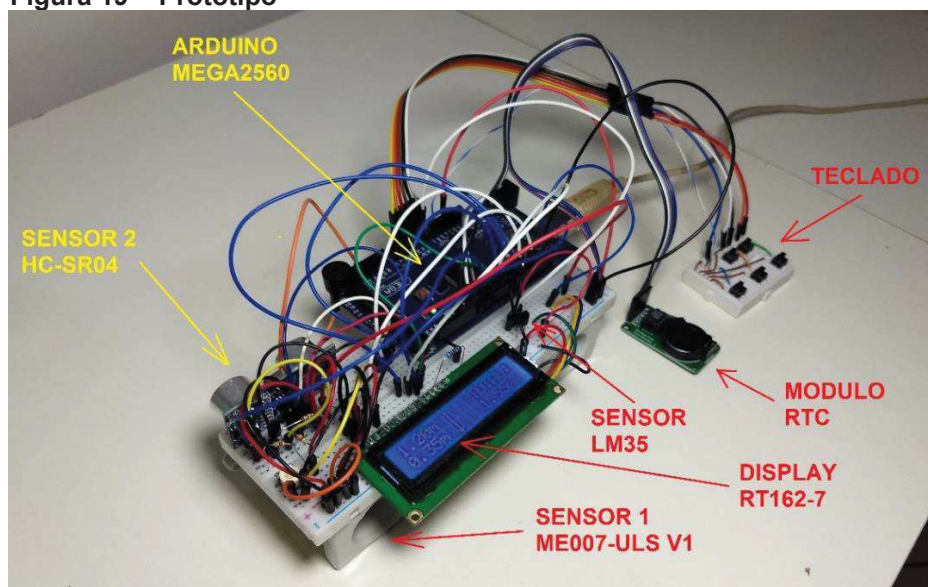
O *display* utilizado foi o modelo RT162-7, que possui dezesseis colunas por duas linhas (16x2). É um *display* de cristal líquido, com um fundo azul e apresenta os caracteres na cor branca.

3.6 Protótipo

O protótipo foi construído com o intuito de se realizar testes de funcionamento e das medidas de forma gradual, ou seja, a cada componente conectado realizava-se a sua implementação no código do programa e os devidos testes de funcionalidade.

Sua construção foi realizada em duas *protoboards*, sendo uma para conexões dos sensores, circuito elevador de tensão, *display* e Arduino, e a outra *protoboard* para conexões de cinco botões com o Arduino, com a disposição conforme pode ser visto na Figura 19.

Figura 19 – Protótipo



Fonte: Autoria própria (2019).

Todos os sensores e módulos utilizados neste projeto podem ser vistos, as suas imagens, no ANEXO B.

Foram desenvolvidas algumas funções com navegação e execução por meio de menus, com a utilização de cinco botões.

Quando o instrumento é ligado, apresenta em seu *display* a data e a hora atuais. Ao pressionar o botão de cima, segue para a próxima função que é a de medição, onde é apresentada a distância acima do instrumento, abaixo e a distância total com a soma das duas distâncias, tudo na mesma tela. Ainda nesta função, também pode-se clicar no botão “varrer”, que realiza uma varredura com várias medições em alguns segundos para encontrar a menor distância e que ao final emite um sinal sonoro (bipe).

Tanto na medição instantânea quanto após a varredura, pode-se clicar no botão “gravar” para que seja armazenado um registro contendo data e hora, distância total e temperatura. Ao clicar novamente no botão para cima, entra-se na função de temperatura, onde é apresentada no *display* a temperatura ambiente atual. Mais uma vez clicando no mesmo botão, vai para a última função que é de consulta de registro. Nesta última, é apresentado o último registro gravado e pode-se clicar no botão esquerdo para consultar registros anteriores e o direito para consultar registros mais recentes.

O menu foi implementado de maneira que se navegue entre as funções principais de forma cíclica, tanto para cima quanto para baixo. A Figura 20 demonstra as disposições das funções.

Figura 20 – Navegação entre funções



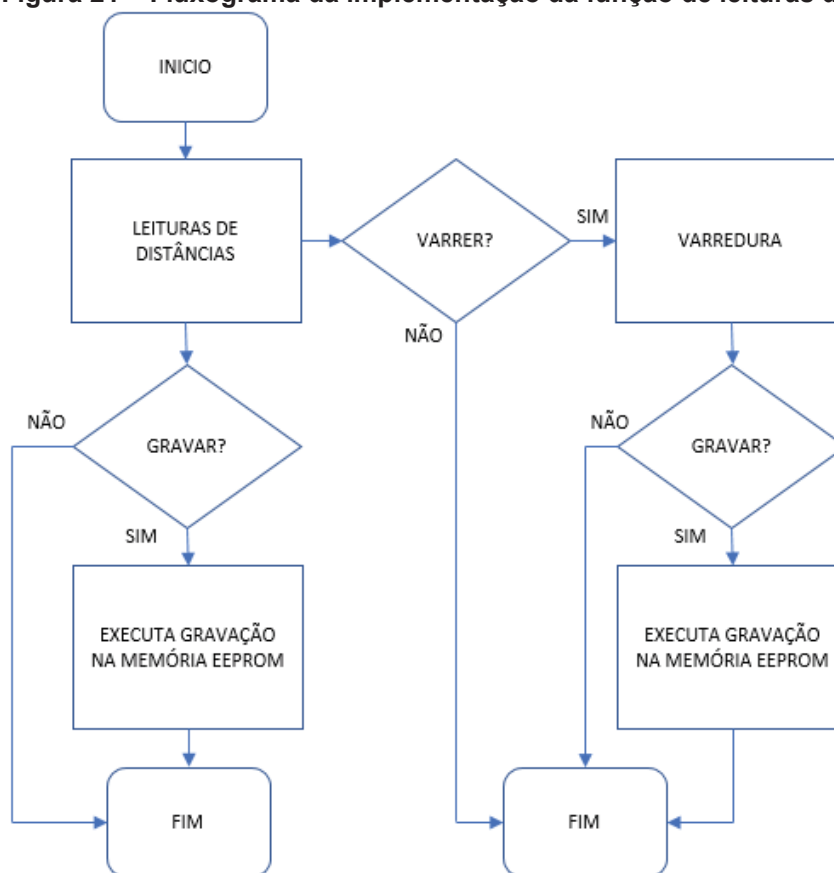
Fonte: Autoria própria (2019).

A Figura 21 detalha através de um fluxograma a implementação da função de leituras das distâncias.

O medidor possui alimentação por meio de uma bateria de 9 V, que se conecta a uma placa desenvolvida para regular a tensão em 5 V. Esta placa alimenta todos os sensores, módulo RTC e Arduino. Nesta placa, há um LED para sinalizar quando o aparelho está ligado ou desligado, acomoda um *buzzer*, o sensor LM35 e uma chave liga-desliga.

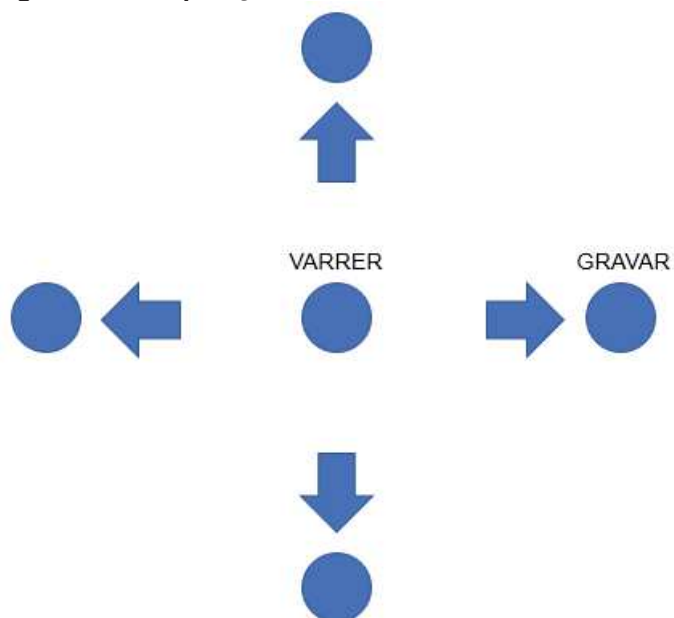
Outra placa que foi desenvolvida é a placa para os botões de navegação entre menus e funções. A Figura 22 ilustra a disposição dos botões (círculos), as direções de navegação (setas) e funções gravar e varrer.

Figura 21 – Fluxograma da implementação da função de leituras das distâncias



Fonte: Autoria própria (2019).

Figura 22 – Disposição do teclado



Fonte: Autoria própria (2019).

4 RESULTADOS E DISCUSSÕES

Conforme mencionado no subitem 3.3.1, Sensores ultrassônicos, somente após entendimento do funcionamento e de como identificar o sinal digital transmitido pelo sensor ME007-ULA V1, foi possível implementar um código de programa no Arduino para realizar a aquisição do sinal e processá-lo para apresentar as informações referentes a cada byte no monitor serial da IDE do Arduino, conforme a Figura 23.

Figura 23 – Apresentação dos bytes em formato decimal no serial monitor da IDE do Arduino

The screenshot shows the Serial Monitor window for COM3. The output displays a sequence of received bytes and their interpretation as frame components. The data is organized into three frames (QUADRO 1, QUADRO 2, and QUADRO 3). Frame 1 contains 7 bytes: 255 (header), 1 (sensor address), 7 (DATA_H), 35 (DATA_L), 1 (TEMP_H), 43 (TEMP_L), and 86 (checksum). Frame 2 contains 7 bytes: 255 (header), 1 (sensor address), 7 (DATA_H), 31 (DATA_L), 1 (TEMP_H), 43 (TEMP_L), and 82 (checksum). Frame 3 contains 7 bytes: 255 (header), 1 (sensor address), 7 (DATA_H), 31 (DATA_L), 1 (TEMP_H), 43 (TEMP_L), and 82 (checksum). The window also shows a 'Send' button, 'Autoscroll' and 'Show timestamp' checkboxes, and a dropdown menu for 'Newline' and '9600 baud'.

```

I received: 255
I received: 1
I received: 7
I received: 35
I received: 1
I received: 43
I received: 86
I received: 255 — BYTE 1: CABEÇARIO DE IDENTIFICAÇÃO DO QUADRO
I received: 1 — BYTE 2: ENDEREÇO DO SENSOR
I received: 7 — BYTE 3: DATA_H
I received: 35 — BYTE 4: DATA_L
I received: 1 — BYTE 5: TEMP_H
I received: 43 — BYTE 6: TEMP_L
I received: 86 — BYTE 7: SOMA DE VERIFICAÇÃO
I received: 255
I received: 1
I received: 7
I received: 31
I received: 1
I received: 43
I received: 82
I received: 255
  
```

Fonte: Autoria própria (2019).

Com a certificação de que as leituras do sensor estavam identificando corretamente cada byte, foram implementadas no código do programa as Equações 2 e 3, do capítulo anterior, para se obter os valores de distância e temperatura.

Em um laboratório, em cima de uma bancada e com o sensor ultrassônico ME007-ULA V1 direcionado para o teto (laje), foram realizados alguns testes de leituras, com os resultados conforme o Quadro 6, a seguir.

Quadro 6 – Resultados de distâncias e temperaturas

| ME007-ULA V1 | | Temp. C° com termom. digital |
|---------------------|-------------------------|-------------------------------------|
| Medida (mm) | Temperatura (C°) | |
| 1834 | 29,1 | 25,9 |
| 1830 | 29,1 | 25,9 |
| 1833 | 29,1 | 25,9 |
| 1833 | 29,3 | 25,9 |
| 1835 | 29,3 | 25,9 |

Fonte: Autoria própria (2019).

Utilizou-se uma trena para comparar a distância medida pelo sensor com a distância física real, o que se verificou estarem iguais. Já na temperatura, os valores apresentados pelo sensor ME007-ULA V1 foram comparados com um termômetro digital, modelo WT-1, da marca Harald, onde foi verificado uma diferença de mais de três graus e que motivou a implementação de um sensor de temperatura independente do sensor ultrassônico ME007-ULA V1.

Foi construído o primeiro protótipo utilizando os sensores ultrassônicos ME007-ULA V1 (com o conversor MAX3232) e o HC-SR04, o sensor de temperatura LM35 e o LCD. Foi realizada uma nova comparação de temperatura, e desta vez, com o resultado obtido do sensor LM35 já no LCD e ao lado o termômetro digital. Foi percebido uma boa melhora na medição da temperatura, pois com o LM35 foi obtido a leitura de 26,39 °C enquanto o termômetro digital apresentava 26,0 °C, onde a diferença ficou em menos de 1 °C.

Com os dois sensores ultrassônicos montados e implementados, foram realizados testes dentro do laboratório posicionando o aparelho entre obstáculos, utilizando pequenas distâncias em cima da própria bancada para verificar se as distâncias de cada sensor ultrassônico estavam corretas, e também a soma das duas distâncias.

Após confirmado que estava tudo funcionando corretamente, foram realizados testes externos para verificação de distâncias maiores.

A céu aberto o sensor ME007-ULA V1 não apresenta valores; quando mirado para o beiral da casa (sobrado), o mesmo registrou 6,5 m de altura. Este valor foi o resultado da soma da altura do aparelho ao chão com a altura do aparelho em relação ao objeto acima.

Quando realizado a medição embaixo do cabo da rede elétrica, o cabo não foi detectado, sendo necessário elevar o dispositivo para uma altura próxima de dois

metros de altura do solo para detectar os cabos de telefonia, que estavam em torno de 50 cm abaixo dos cabos de baixa tensão, onde foi obtida a medida de 4,98 m. Esta altura dos cabos de telefonia até o chão foi medida com o uso de uma vara telescópica de medição de alta isolação, concedida por empréstimo pela empresa Copel. A medição do aparelho comparando com a medição da vara telescópica resultou na diferença de poucos milímetros, tendo em vista que a vara possui uma régua com medidas em centímetros.

No laboratório da universidade foram realizados vários testes. Os testes consistiram em verificar a detecção de um cabo de energia com 1 cm de diâmetro, que estava preso no teto do laboratório e perpendicularmente chegava quase ao chão, no meio da sala.

As medições foram feitas na horizontal, sendo de uma parede até o cabo, com o aparelho posicionado a um metro da parede. Os resultados obtidos foram: do dispositivo até o cabo, 2,63 m; do dispositivo até um cabo de madeira (de aproximadamente 2 cm), obteve-se 3,50 m; até uma barra de alumínio, de 5 cm de largura e 3 cm de profundidade, obteve-se 4,10 m.

As medidas eram obtidas/atualizadas no display a cada 500 ms, pois esse era o ciclo de leitura *default* do ME007-ULA V1, então para um objeto bastante fino tinha-se que segurar o medidor o mais estático possível, pois o foco da direção do sensor é uma faixa com ângulo de abertura de 8° , exigindo uma boa mira do operador.

Foi substituído o sensor ultrassônico ME007-ULA V1 pelo ME007-ULS V1, pois este último é um modelo que possui o ciclo de leitura mais rápido, determinado por um gatilho, que pode ser disparado gerando ciclos de leitura com o mínimo de 60 ms de intervalo. Com isto, foi possível realizar detecções mais rápidas, facilitando o posicionamento do aparelho para detectar o objeto, apresentando a informação mais instantaneamente.

Novamente no laboratório, mesmo local dos testes realizados com o outro sensor ultrassônico, o ME007-ULA V1, foram repetidos os testes realizados anteriormente, mas desta vez, com o ME007-ULS-V1.

O segundo sensor apresentou o mesmo comportamento do anterior com mesmas medidas, mas após realizar as medidas foi balançado o cabo, como um pêndulo, então apareceram novas detecções com distâncias um pouco maiores, de aproximadamente 50 cm e variava. Diante disso, foram realizados novos testes em campo aberto, no mesmo local que foram feitos os primeiros testes.

O protótipo implementado com o sensor ME007-ULS-V1 conseguiu realizar a leitura da distância dos cabos telefônicos, em altura confortável, ou seja, houve um aumento da distância detectada pelo sensor ME007-ULS-V1 e diminuindo a do sensor HC-SR04. A Figura 24 apresenta o resultado obtido no LCD.

Figura 24 – Resultado de medidas de cabos telefônicos em campo aberto



Fonte: Autoria própria (2019).

A distância de 3,61 m informada na Figura acima, foi a máxima distância detectável pelo sensor ME007-ULS-V1, onde deve-se considerar que haviam oito cabos juntos e que os cabos telefônicos eram do tipo *drop*, que possuem largura de mais ou menos 6 mm cada.

Foi realizado teste em outro local onde haviam somente cabos de rede de distribuição de baixa tensão, onde não foi obtido sucesso na leitura. Verificado com a vara telescópica de medição que a distância era de 5,54 m. Os cabos possuíam a bitola de 2/0 AWG (67,35 mm²), conforme informado pelo técnico da Copel.

Também foi realizado um teste de distância entre dois muros paralelos. Estes muros possuíam dois metros de altura e comprimento de cinco metros cada. Com uma trena, foi medida a distância interna de um muro ao outro, onde se obteve a distância de 9,75 m. O aparelho conseguiu realizar a medição com bastante precisão, conforme apresentado na Figura 25.

Figura 25 – Resultado de medidas entre dois muros



Fonte: Autoria própria (2019).

Cabe ressaltar que as medidas obtidas a partir das leituras dos sensores ultrassônicos em diferentes ocasiões e locais, se apresentaram bastante exatas. Isso se deve ao fato de que os sensores possuem compensação de temperatura para que se possa ter um cálculo da velocidade do som, e conseqüentemente a distância, com maior precisão.

A próxima etapa após a realização dos testes de distancias e temperatura foi da implementação de um menu de funções, navegável por um teclado com cinco botões.

O medidor foi implementado com mais duas funções, uma função que apresenta a data e o horário atual e outra função que realiza a consulta de registros gravados na memória. Todo o código da programação pode ser verificado no ANEXO D.

Em primeiro, ao ligar o aparelho, é a apresentada a tela de data e horas, programado com a disposição que pode ser visto na Figura 26.

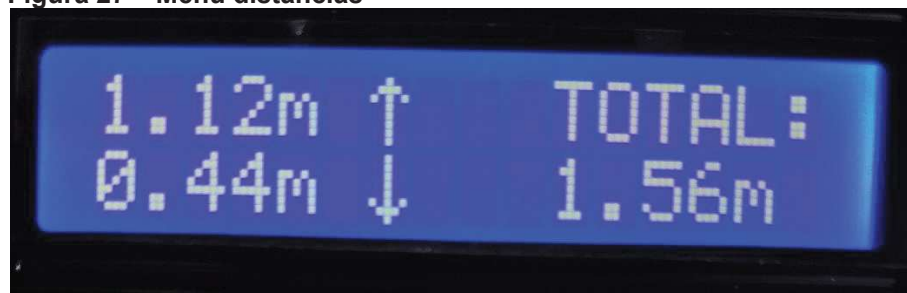
Figura 26– Menu data e hora



Fonte: Autoria própria (2019).

Na sequência, é apresentada a próxima tela de menu com as informações de distâncias para do aparelho para cima, do aparelho para baixo e distancia total com a soma das duas anteriores. A Figura 27 apresenta como ficou.

Figura 27 – Menu distâncias



Fonte: Autoria própria (2019).

A próxima tela refere-se ao menu de temperatura, apresentando a temperatura ambiente, a qual é apresentada na Figura 28.

Figura 28 – Menu temperatura ambiente



Fonte: Autoria própria (2019).

Por fim, na última tela de menu, é apresentada no *display* o último registro gravado. Este registro traz na tela as informações de data e horário da distância coletada, e também a temperatura no momento. Esta função permite que se possa consultar registros anteriores. A seguir, a Figura 29 apresenta esta tela.

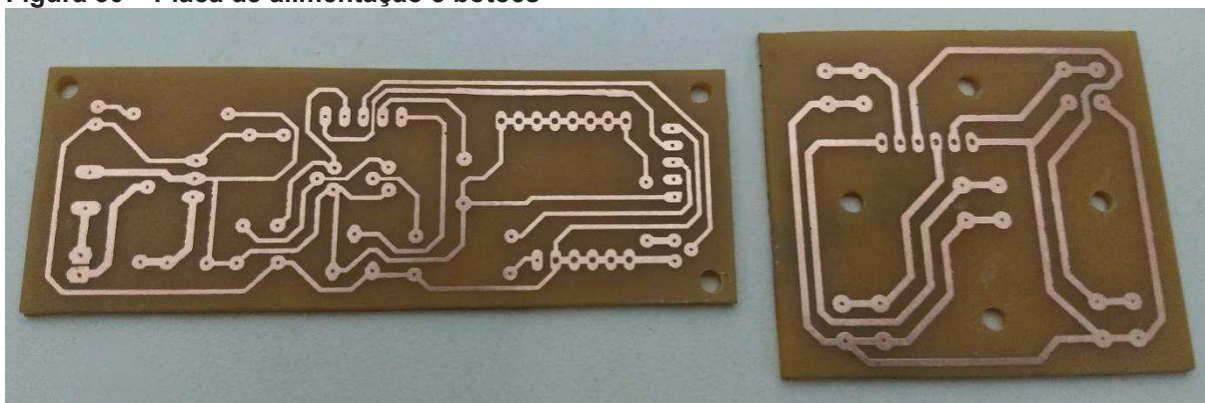
Figura 29 – Menu consulta registro



Fonte: Autoria própria (2019).

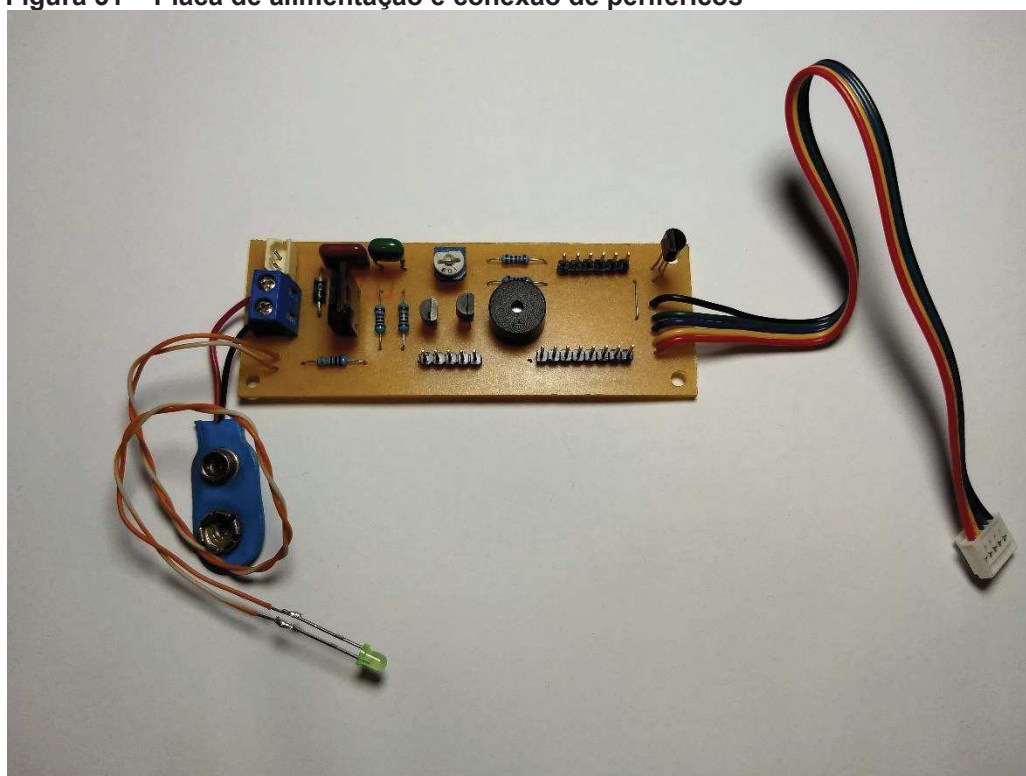
Abaixo, as Figuras 30, 31 e 32 apresentam como ficaram as placas desenvolvidas e construídas para alimentação, regulação de tensão, conexão dos sensores e módulos, e botões. Os respectivos circuitos eletrônicos desenvolvidos podem ser verificados no ANEXO A.

Figura 30 – Placa de alimentação e botões

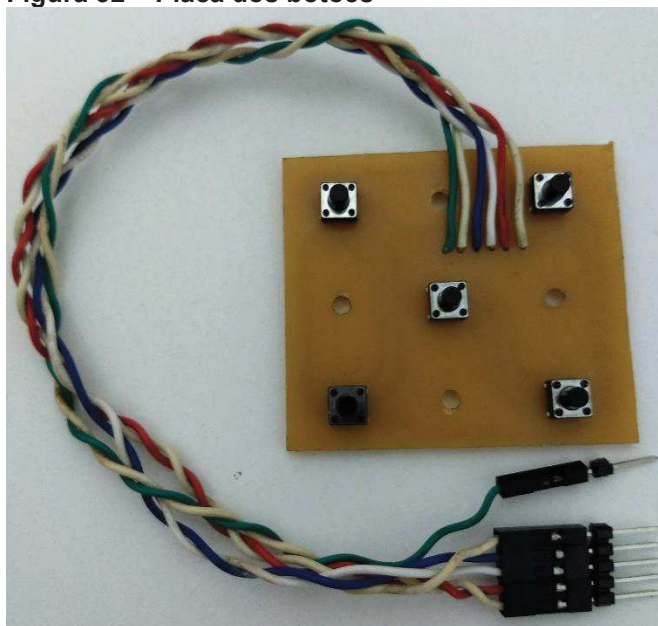


Fonte: Autoria própria (2019).

Figura 31 – Placa de alimentação e conexão de periféricos



Fonte: Autoria própria (2019).

Figura 32 – Placa dos botões

Fonte: Autoria própria (2019).

O projeto da caixa foi desenvolvido observando um bom acondicionamento dos sensores, módulos, placas e bateria, como também, uma boa ergonomia para o operador. Vale ressaltar, que neste aparelho desenvolvido, há um sensor direcionado para baixo, o que contribui para a ergonomia do mesmo dispensando o auxílio de réguas ou colocando-o no chão, que é o caso dos dois modelos comerciais encontrados. A seguir, a Figura 33 mostra como ficou o medidor finalizado.

Figura 33 – Medidor finalizado



Fonte: Autoria própria (2019).

5 CONCLUSÃO

Neste trabalho foi realizado o desenvolvimento de um dispositivo que realiza a medição das distâncias através da tecnologia ultrassônica e que também a medição da temperatura ambiente em graus Celsius.

Neste instrumento, além da medição de distâncias e temperatura ambiente, é possível consultar a data, hora, minutos e segundos atuais. É possível navegar nestas opções de consultas e também realizar a gravação destas informações na memória do aparelho. Também pode-se consultar os registros gravados anteriormente em horários e datas diferentes, dispensando a utilização de lápis ou caneta e papel para anotações e possibilitando verificação posteriormente no escritório ou em qualquer outro lugar.

Verificou-se que os resultados obtidos das medidas das distâncias realizadas não foram satisfatórios, tendo vista que, para cabos de média tensão seria necessário a detecção de medidas de pelo menos sete metros, conforme indicado no item 2.2 no quesito de distâncias mínimas. Foi possível somente medir a altura da rede de telefonia, que está a uma distância um pouco mais abaixo das redes de baixa tensão e média tensão.

Conforme testes realizados, foi percebido que quanto maior a distância do cabo a ser detectado, menor deve ser a resolução do sensor ultrassônico. Apesar do modelo de sensor escolhido possuir uma resolução de 1 mm, não foi o suficiente para chegar nas medidas mínimas de altura dos cabos de rede para utilização profissional. Para isso, seria necessário utilização de um sensor ultrassônico mais elaborado, com resolução inferior a 1 mm, o que reflete em um custo maior. O ANEXO C apresenta um orçamento de dois sensores ultrassônicos com resolução e precisão mais elaboradas.

Para se manter um melhor posicionamento do aparelho pode ser implementado um nível tipo bolha redondo, para se obter uma medida mais próxima da vertical possível.

Apesar de não ter sido atingido o objetivo principal com medição satisfatória de cabos de redes, o medidor desenvolvido pode ser utilizado em outros tipos de medições, tendo em vista que em testes realizados com objetos grandes chegou a medida de 9,75 m. Pode ser utilizado como por exemplo, na medição de uma laje, na

medição de cômodos, e outras aplicações onde os objetos não sejam tão estreitos como um cabo.

REFERÊNCIAS

ALIEXPRESS. **Digital lcd ultrasonic cable height meter handheld height cable tester measuring the height of overhead cables 3-23m.** Disponível em: <<https://www.aliexpress.com/item/32846346114.html?spm=a2g0o.detail.1000016.1.17c41c35Hwa403&isOrigTitle=true>>. Acesso em: 8 ago. 2018a.

ALIEXPRESS. **Ultrasonic RS232 Output/Analog Output Sensor Small Angle Long Distance Ranging Module.** Disponível em: <<https://pt.aliexpress.com/item/32801031531.html?spm=a2g0s.9042311.0.0.3da24c4d8UfBpn>>. Acesso em: 20 nov. 2018b.

ALIEXPRESS. **With temperature compensated ultrasonic / three outputs, small angle remote ranging module can be selected.** Disponível em: <<https://pt.aliexpress.com/item/32429955991.html>> Acesso em: 20 mar. 2019.

ANEEL. **Procedimentos de Distribuição de Energia Elétrica no Sistema Elétrico Nacional.** Disponível em: <http://www.aneel.gov.br/documents/656827/14866914/PRODIST-M%C3%B3dulo3_Revis%C3%A3o7/ebfa9546-09c2-4fe5-a5a2-ac8430cbca99>. Acesso em: 21 out. 2018a.

ANEEL. **Regulação dos Serviços de Distribuição.** Disponível em: <<http://www.aneel.gov.br/regulacao-dos-servicos-de-distribuicao>>. Acesso em: 14 out. 2018b.

AUTOMATIZE. **Sensores Ultra-sônicos Siemens.** 2018. Disponível em: <<http://www.automatizesensores.com.br/ultrasonicos.html>>. Acesso em: 8 ago. 2018.

COPEL. **NTC 810 008: CABOS DE ALUMÍNIO NU TIPOS CA e CAA.** Curitiba: Copel, 2018a. Disponível em: <[https://www.copel.com/hpcopel/normas/ntcarquivos.nsf/236CB379AEA9704F83258257005E22CE/\\$FILE/810008%20_%20cabos%20al%20nus%20CA%20e%20CAA_mar2018.pdf](https://www.copel.com/hpcopel/normas/ntcarquivos.nsf/236CB379AEA9704F83258257005E22CE/$FILE/810008%20_%20cabos%20al%20nus%20CA%20e%20CAA_mar2018.pdf)>. Acesso em: 30 mai. 2019.

COPEL. **NTC 850 001: DIMENSIONAMENTO DE ESTRUTURAS.** Curitiba: Copel, 2015. Disponível em: <[https://www.copel.com/hpcopel/normas/ntcarquivos.nsf/4BA228451FDED5CA03257E2300596969/\\$FILE/Ntc%20Dimens%20-%20Jul95.pdf](https://www.copel.com/hpcopel/normas/ntcarquivos.nsf/4BA228451FDED5CA03257E2300596969/$FILE/Ntc%20Dimens%20-%20Jul95.pdf)>. Acesso em: 12 out. 2018.

COPEL. **NTC 850 005**: TABELAS PARA PROJETO E MONTAGEM DE LINHAS E REDES DE DISTRIBUIÇÃO. Curitiba: Copel, 2018b. Disponível em: <[https://www.copel.com/hpcopel/normas/ntcarquivos.nsf/83A4327EB60EBB7203257E230059A60C/\\$FILE/Ntc%20850005%20-%20Abr96.pdf](https://www.copel.com/hpcopel/normas/ntcarquivos.nsf/83A4327EB60EBB7203257E230059A60C/$FILE/Ntc%20850005%20-%20Abr96.pdf)>. Acesso em: 12 out. 2018.

COPEL. **NTC 855 901**: COMPARTILHAMENTO DE INFRA-ESTRUTURA DE REDES DE DISTRIBUIÇÃO. Curitiba: Copel, 2018c. Disponível em: <[http://www.copel.com/hpcopel/normas/ntcarquivos.nsf/29E1E02520B3E7C3832583070061BDE7/\\$FILE/NTC%20855901%20Compartilhamento%20Setembro%202018.pdf](http://www.copel.com/hpcopel/normas/ntcarquivos.nsf/29E1E02520B3E7C3832583070061BDE7/$FILE/NTC%20855901%20Compartilhamento%20Setembro%202018.pdf)>. Acesso em: 12 out. 2018.

COPEL. **NTC 856 004**: Afastamentos mínimos entre condutores e solo. Curitiba: Copel, 2016a. Disponível em: <[http://www.copel.com/hpcopel/normas/ntcArquivos.nsf/7C861BFED6C66A9483258092003FCFF9/\\$FILE/RDA%20856004%20Primario%20Afastamentos%20minimos%20entre%20condutores%20e%20solo%20dezembro2016.pdf](http://www.copel.com/hpcopel/normas/ntcArquivos.nsf/7C861BFED6C66A9483258092003FCFF9/$FILE/RDA%20856004%20Primario%20Afastamentos%20minimos%20entre%20condutores%20e%20solo%20dezembro2016.pdf)>. Acesso em: 12 out. 2018.

COPEL. **NTC 856 116**: MONTAGEM DE REDES DE DISTRIBUIÇÃO AÉREA. Curitiba: Copel, 2016b. Disponível em: <[https://www.copel.com/hpcopel/normas/ntcArquivos.nsf/4BA78ECC0DBCDFEB03257F3E006079F4/\\$FILE/RDA%20856114%20Primario%20N3%20janeiro2016.pdf](https://www.copel.com/hpcopel/normas/ntcArquivos.nsf/4BA78ECC0DBCDFEB03257F3E006079F4/$FILE/RDA%20856114%20Primario%20N3%20janeiro2016.pdf)>. Acesso em: 30 mai. 2019.

COSTA, Leonardo Faria. **Sensores industriais: Sensor ultrassônico**. 08 ago. 2016, 16 dez. 2016. Notas de Aula.

D'ASSIS. **Vara de Manobra Telescópica Medição**. 2018. Disponível em: <www.lojavirtual.dassis.com.br/va>. Acesso em: 20 out. 2018.

DOCPLAYER. **Ultrasonic Ranging Controller**. Disponível em: <<https://docplayer.net/52320401-Ultrasonic-ranging-controller.html>>. Acesso em: 04 abr. 2019.

ELETROBRAS. **Como a energia elétrica é gerada no Brasil**. Disponível em: <<http://www.eletrobras.com/elb/natrilhadaenergia/energia-eletrica/services/eletrobras/trilhaenergia/pdfs/como-a-energia-eletrica-e-gerada-no-brasil.pdf>>. Acesso em: 14 out. 2018. 2018a.

ELETROBRAS. **Como a energia elétrica é transmitida no Brasil**. Disponível em: <<http://www.eletrobras.com/elb/natrilhadaenergia/energia-eletrica/services/eletrobras/trilhaenergia/pdfs/como-a-energia-eletrica-e-transmitida-no-brasil.pdf>>. Acesso em: 14 out. 2018. 2018b.

ELETROBRAS. **Importância da energia elétrica**. Disponível em: <<http://www.eletrobras.com/elb/natrilhadaenergia/energia-eletrica/services/eletrobras/trilhaenergia/pdfs/importancia-da-energia-eletrica.pdf>>. Acesso em: 14 out. 2018. 2018c.

ESCOLA, Brasil. **CONSTRUINDO UM TEODOLITO**. 2018. Disponível em: <<https://educador.brasilecola.uol.com.br/estrategias-ensino/construindo-um-teodolito.htm>>. Acesso em: 20 out. 2018.

FREITAS, Stefani. **Geração, transmissão e distribuição de energia elétrica**. s/d. Notas de Aula.

PHILODENDROS. **Equipamentos Mensuração da Distância**. 2018. Disponível em: <http://cmq.esalq.usp.br/Philodendros/doku.php?id=equipamentos:distancia:distancia_analogicos>. Acesso em: 20 out. 2018.

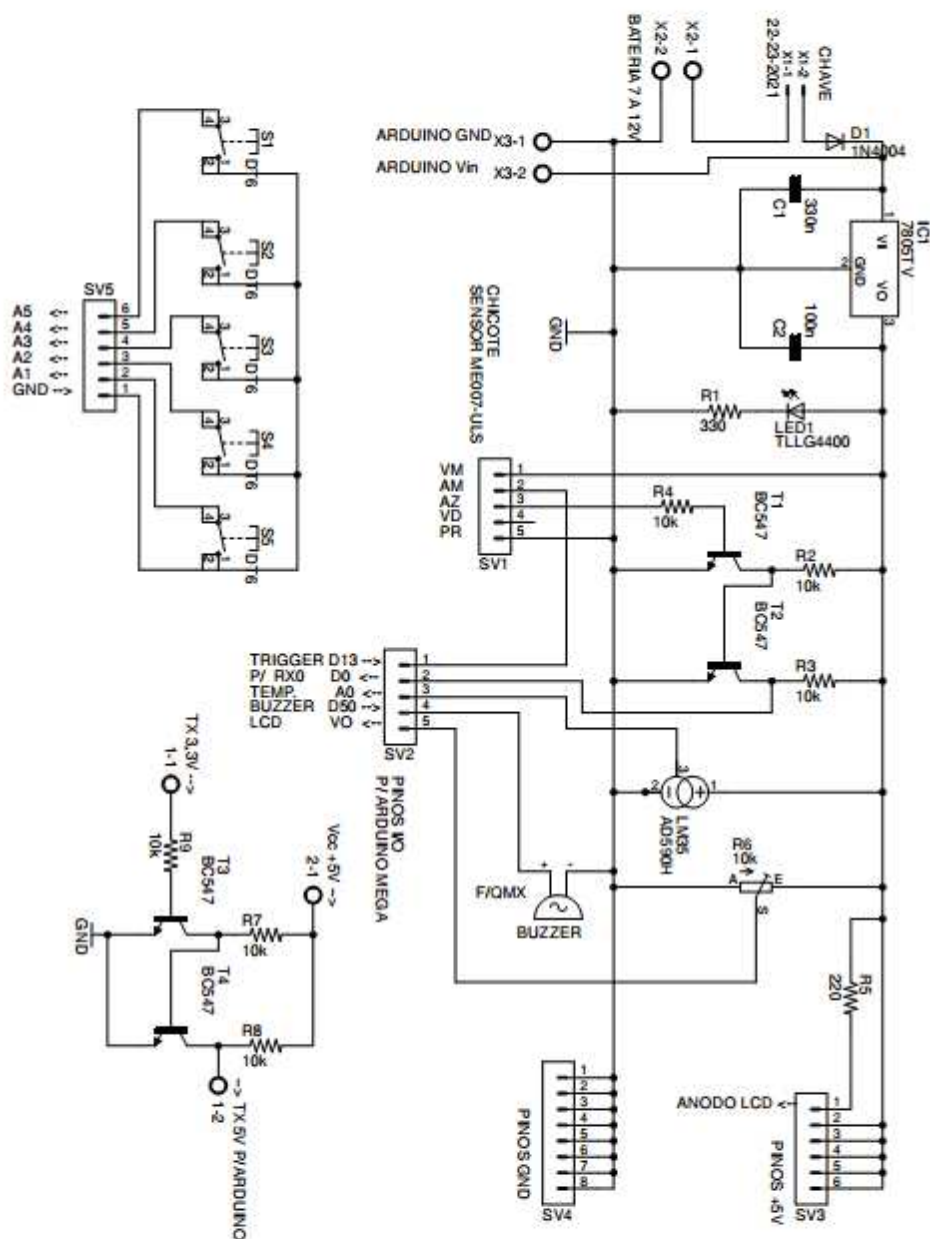
RITZMG. **Vara de Manobra Telescópica**. Disponível em: <<http://www.ritzmg.com.br/site/vmr-45---vara-de-manobra-telesc-pica>> Acesso em: 7 jun. 2019.

THOMAZINI, Daniel; ALBUQUERQUE, Pedro Urbano Braga de. **Sensores Industriais: Fundamentos e Aplicações**. 8. ed. São Paulo: Érica, 2012.

ZENITH SURVEY. **SUPARULE CHM300 CABLE HEIGHT METER**. Disponível em: <<https://www.zenithsurvey.co.uk/product/suparule-chm300-cable-height-meter/>>. Acesso em: 8 ago. 2018.

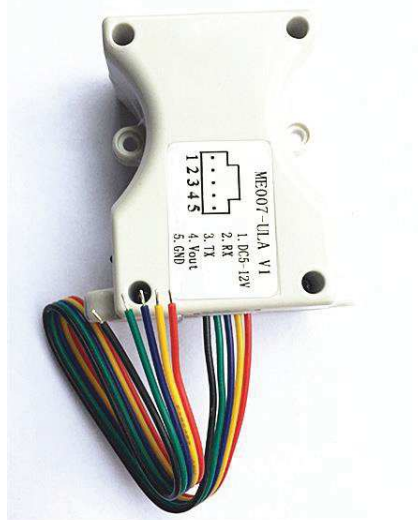
ANEXOS

ANEXO A – Circuitos desenvolvidos e utilizados



ANEXO B – Imagens dos sensores e módulos utilizados

SENSOR ULTRASSÔNICO ME007-ULA V1:



FONTE: ALIEXPRESS (2018).

SENSOR ULTRASSÔNICO ME007-ULS V1:



FONTE: ALIEXPRESS (2019).

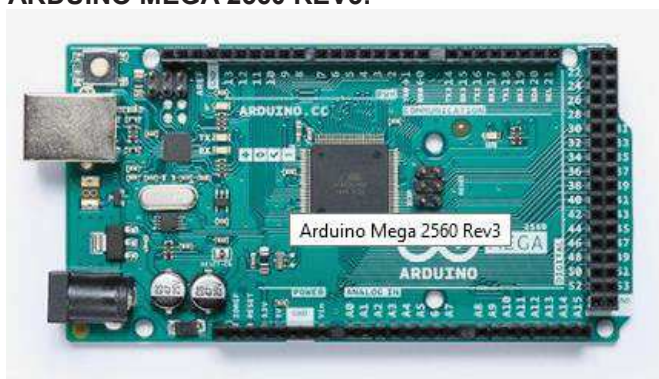
SENSOR ULTRASSÔNICO HC-SR04:



FONTE: ALIEXPRESS (2019).

SENSOR DE TEMPERATURA LM35:

FONTE: FILIPFLOP (2019).

ARDUINO MEGA 2560 REV3:

FONTE: ARDUINO (2019).

MÓDULO RTC DS1302:

FONTE: ALIEXPRESS (2019).

MÓDULO LCD RT162-7:

FONTE: ALIEXPRESS (2019).

ANEXO C – Orçamento de dois modelos de sensores ultrassônicos solicitados à empresa SENSE.



| PROPOSTA Nº CATC 1900508 | | Data: 08/05/2019 | | F 1/2 | | | |
|---|--|--|---|-----------------|-----|-----------------|-----------------|
| Para : | Carlos Eduardo Porto de Almeida Schork | De: | ELCONI COM. E REPR. DE MAT. | | | | |
| Email | carlooseduardotsi@yahoo.com.br | Email: | vendas@elconi.com.br - clovis@elconi.com.br Maria José Shtorache - Clóvis Carvalho | | | | |
| Depto: | | Depto: | Comercial | | | | |
| Fone | (41) 9999.99999 | Fone/Fax: | (41) 3352-3022 | | | | |
| Conforme solicitação , apresentamos proposta para fornecimento dos seguintes itens: | | | | | | | |
| ITEM | QUANT | ESPECIFICAÇÃO | P. UNI. | TOTAL S/ IPI | IPI | IPI R\$ | TOTAL C/ IPI |
| 1 | 1 | Sensor ultrassônico digital M30-6m, PNP, conector M12-5p Sense MIC+600/D/TC - PN 5000002681 - NCM 8536.50.90 | 3.200,00 | 3.200,00 | 15% | 480,00 | 3680,00 |
| 2 | 1 | Sensor ultrassônico analógico M30-6m, 0-10V/4-20mA, conector Sense MIC+600/IU/TC - PN 5000002682 - NCM 8536.50.90 | 5.500,00 | 5.500,00 | 15% | 825,00 | 6325,00 |
| | | | | - | | - | 0,00 |
| | | | | - | | - | 0,00 |
| | | | | - | | - | 0,00 |
| | | | | - | | - | 0,00 |
| Valor da Proposta sem ST | | | | 8.700,00 | | 1.305,00 | 10005,00 |

ANEXO D – Código de programa implementado na IDE do ARDUINO

```
//MEDIDOR ULTRASSÔNICO DE DISTÂNCIA PARA APLICAÇÃO EM CONDUTORES DE DISTRIBUIÇÃO DE ENERGIA
ELÉTRICA

//Trabalho de Conclusão de Curso - Engenharia Eletrônica
//Carlos Eduardo Porto de Almeida Schork

//-----Bibliotecas-----
#include <virtuabotixRTC.h>
#include <stdio.h>
#include <DS1302.h>
#include <Ultrasonic.h>
#include <LiquidCrystal.h>
#include <SoftwareSerial.h>
#include <ctype.h>
#include <EEPROM.h>

//-----Mapeamento de Hardware-----
//#define espacoEEPROM 1024 //1Kb para todos os Arduinos com ATmega328P
#define pinTrigger 13 //define pino 13 para gerar o trigger no sensor ME007

#define botUp A1 //Botão para selecionar tela acima no pino A1
#define botDown A2 //Botão para selecionar tela abaixo no pino A2
#define botEsq A3 //Botão para sair da tela
#define botDirGravar A4 //Botão para entrar na tela ou gravar
#define botVarrer A5 //Botão para pausar a medicaçao ou gravar a leitura

#define clk 14 //Pino CLK do RTC DS1302
#define dat 15 //Pino DAT do RTC DS1302
#define rst 16 //Pino RST do RTC DS1302

#define buzzer 50 //pino do buzzer

//-----Decaracao de objetos-----
Ultrasonic ultrassom(8,7); //define o nome do sensor HC-SR04 (ultrassom)e onde esta ligado o
trig(8) e o echo(7) respectivamente
LiquidCrystal lcd(12, 11, 5, 4, 3, 2); //Define os pinos que serão utilizados para ligação ao
display
virtuabotixRTC myRTC(clk, dat, rst); //declara objeto para o RTC

//-----Variaveis globais-----
float temperatura;

String bkpTime; //Variavel para armazenar data e hora em formato string
String bkpGeral; //Variavel para armazenar data e hora, distancia e temperatura em
formato string

const int LM35 = A0; //Define o pino analogico que lera o LM35

char menu = 0x01; //Variável para selecionar o menu
boolean t_botUp; //Flags para armazenar o estado dos botões
boolean t_botDown;
boolean t_botEsq;
boolean t_botDirGravar;
boolean t_botVarrer;
int count;
int countVarrer;
int valorUltimaGravacao;
int eeAdress = 0;
int adress = 0;
float distancia1; //Mede distancia para cima do dispositivo
float distancia2; //Mede distancia para baixo do dispositivo
float distancias[5]; //Array com 5 posições para guardar as medidas da
distancia_total
float menor_distancia_detectada;
float distancia_varrida;
float distancia_varrida_final;
float gravarDistancia;

char set1 = 0x00, set2 = 0x00; //variaveis para controle do case para varredura e da
gravacao

byte grau[8] = { B00001100, //Array simbolo grau
B00010010,
```

```

        B00010010,
        B00001110,
        B00000000,
        B00000000,
        B00000000,
        B00000000,};

byte flechaUp[8]   = { B00000100,           //Array simbolo flecha cima
                      B00001110,
                      B00010101,
                      B10000100,
                      B00000100,
                      B00000100,
                      B00000100,
                      B00000100,};

byte flechaDown[8] = { B00000100,           //Array simbolo flecha baixo
                      B00000100,
                      B00000100,
                      B10000100,
                      B00000100,
                      B00010101,
                      B00001110,
                      B00000100,};

byte  valorIndice;
float valorDistancia;
float valorTemperatura;
byte  valorDiadoMes;
byte  valorMes;
int   valorAno;
byte  valorHoras;
byte  valorMinutos;
byte  valorSegundos;

byte  leIndice;
float leituraDistancia;
float leituraTemperatura;
byte  leDiadoMes;
byte  leMes;
int   leituraAno;
byte  leHoras;
byte  leMinutos;
byte  leSegundos;

int flag_gravou = 0;
int flag_esq = 0;
int registro = EEPROM.read(0);

//-----Declaracao de funcoes-----
void MudaMenu();
void DispMenu();
void TriggersensorME007ULS();
void Distancias();
void Buzzer();
void TemperaturaAmbiente();
void Data_hora();
void ConsultaRegistro();
void Varredura();
void Gravar();
void InformacaoMonitor();
void InformacaoNoLCD();
void EEPROMMonitor();

//=====Configuracao de Inicializacao=====
void setup() {

    Serial.begin(9600);           //Inicia a porta serial e define velocidade de 9600 bps
    lcd.begin(16, 2);            //Define o número de colunas e linhas do LCD
    lcd.createChar(0, grau);     //Cria o caractere customizado com o simbolo do grau
    lcd.createChar(1, flechaUp); //Cria o caractere customizado com o simbolo do flecha
    cima
    lcd.createChar(2, flechaDown); //Cria o caractere customizado com o simbolo do flecha
    baixo

```

```

pinMode(pinTrigger, OUTPUT);           //Define o pino 13 para saida
pinMode(botUp, INPUT);                 //Entrada para os botões (digitais A1 a A5) transformando
em pull-ups
    digitalWrite(botUp,HIGH);
pinMode(botDown, INPUT);
    digitalWrite(botDown,HIGH);
pinMode(botEsq, INPUT);
    digitalWrite(botEsq,HIGH);
pinMode(botDirGravar, INPUT);
    digitalWrite(botDirGravar,HIGH);
pinMode(botVarrer, INPUT);
    digitalWrite(botVarrer,HIGH);

pinMode(buzzer, OUTPUT);

t_botUp          = 0x00;                //limpa flags dos botoes
t_botDown        = 0x00;
t_botEsq         = 0x00;
t_botDirGravar   = 0x00;
t_botVarrer      = 0x00;

count = 0;
countVarrer = 0;
distancia_varrida = 0;

//-----apagar toda memoria EEPROM-----
//   for (int nL = 0; nL < EEPROM.length(); nL++)
//   {
//       EEPROM.write(nL, 0xFF);
//   }
//-----

}

//=====Loop Infinito=====
void loop()
{
    MudaMenu();
    DispMenu();
    //EEPROMMonitor();
}

//=====Funcao Muda Menu=====
void MudaMenu()
{
    if(!digitalRead(botUp)) t_botUp = 0x01;           //Botão Up pressionado? Seta flag
    if(!digitalRead(botDown)) t_botDown = 0x01;       //Botão Down pressionado? Seta flag

    if(digitalRead(botUp) && t_botUp)                 //Botão Up solto e flag setada?
    {                                                  //Sim...
        t_botUp = 0x00;                               //Limpa flag

        lcd.clear();                                 //Limpa display
        menu++;                                       //Incrementa menu

        if(menu > 0x04) menu = 0x01;                 //Se menu maior que 4, volta a ser 1
    }

    if(digitalRead(botDown) && t_botDown)             //Botão Down solto e flag setada?
    {                                                  //Sim...
        t_botDown = 0x00;                             //Limpa flag

        lcd.clear();                                 //Limpa display
        menu--;                                       //Decrementa menu

        if(menu < 0x01) menu = 0x04;                 //Se menu menor que 1, volta a ser 4
    }
}

//=====Funcao DispMenu=====
void DispMenu()                                     //Mostra o menu atual
{
    switch(menu)                                     //Controle da variável menu
    {
        case 0x01:

```

```

        Data_hora(); //Chama função data e hora
    break;

    case 0x02:
        TriggersensorME007ULS(); //Dispara o trigger (gatilho para o
sensor ME007
        Distancias(); //Chama a função das leituras
ultrassonicas

        //-----botao varredura-----

        if(!digitalRead(botVarrer)) t_botVarrer = 0x01; //Botão Varrer pressionado?
Seta flag

        if(digitalRead(botVarrer) && t_botVarrer) //Botão Varrer solto e flag
setada?
        {
            //Sim...
            t_botVarrer = 0x00; //Limpa flag
            lcd.clear();
            Varredura();

            while((flag_gravou == 0))
            {
                if(!digitalRead(botUp)) t_botUp = 0x01; //Botão Up
pressionado? Seta flag
                if(!digitalRead(botDown)) t_botDown = 0x01; //Botão Down
pressionado? Seta flag

                if(digitalRead(botUp) && t_botUp) //Botão Up solto e
flag setada?
                {
                    //Sim...
                    t_botUp = 0x00; //Limpa flag
                    flag_gravou = 1;
                }

                if(digitalRead(botDown) && t_botDown) //Botão Down solto e
flag setada?
                {
                    //Sim...
                    t_botDown = 0x00;
                    flag_gravou = 1;
                }

                if(!digitalRead(botDirGravar)) t_botDirGravar = 0x01; //Botão Gravar
pressionado? Seta flag

                if(digitalRead(botDirGravar) && t_botDirGravar) //Botão Gravar
solto e flag setada?
                {
                    //Sim...
                    t_botDirGravar = 0x00; //Limpa flag
                    flag_gravou = 1;
                    lcd.clear();
                    lcd.setCursor(4, 0);
                    lcd.print("GRAVANDO...");
                    delay(100);
                    gravarDistancia = distancia_varrida_final; //atribui resultado a
variavel para gravar
                    Gravar();
                }

            }
            flag_gravou = 0;
        } //end botVarrer

    break;

    case 0x03:
        TemperaturaAmbiente(); //Chama a função temperatura ambiente
    break;

    case 0x04:
        ConsultaRegistro(); //Chama função ConsultaRegistro
    break;

```

```

    }
}
//=====Funcao trigger=====
void TriggersensorME007ULS()
{
    digitalWrite(pinTrigger,HIGH);    //Define o pino 13 para nivel  alto
    delay(60);                        //Espera por 60ms
    digitalWrite(pinTrigger,LOW);     //Define o pino 13 para nivel  baixo
    delay(1);                          //Espera por 1ms
    digitalWrite(pinTrigger,HIGH);    //Define o pino 13 para nivel  alto
}

//=====Funcao Distancias=====
void Distancias()
{
//-----sensor ultrassom 1- ME007ULS-----
int meuArray[6];                      //Array com 6 posições para guardar os bytes do sensor
ME007

int cabecalho;
int endereco;
int dist_H;
int dist_L;
int temp_H;
int temp_L;
int checksum;
float altura_up;
float distancia_total;                //Soma das distancias para cima e para baixo

while(Serial.available()<6) {}        //Aguarda ate que sejam recebidos 6 bytes
    for(int n=0; n<6; n++)
        meuArray[n] = Serial.read(); //Pega os bytes

        cabecalho = meuArray[0];      //Atribui cada posicao do array a uma variavel
        dist_H    = meuArray[1];
        dist_L    = meuArray[2];
        temp_H    = meuArray[3];
        temp_L    = meuArray[4];
        checksum  = meuArray[5];

        altura_up = ((dist_H * 256) + dist_L); //Calcula a distancia a partir dos dois bytes
dist com resultado em mm.
        distancia1 = (altura_up + 55) / 1000;    //Convertendo para metros

//-----sensor ultrassom 2-----

        float altura_down;
        altura_down = ultrassom.Ranging(CM);
        distancia2 = altura_down / 100;        //Convertendo para metros

//-----somando as duas distancias-----

        distancia_total = distancia1 + distancia2;

//-----coletando menor distancia a cada 5 leituras do sensor-----
        if (distancia1 != 0)                //somente medidas do sensor 1 (ME007)
diferente de zero
        {
            distancias[count] = distancia_total;
            count++;
        }

        if (count == 5)
        {
            count = 0;

            if ( (distancias[4] < distancias[3]) && (distancias[4] < distancias[2]) &&
(distancias[4] < distancias[1]) && (distancias[4] < distancias[0]) )
                menor_distancia_detectada = distancias[4];

```



```

        if ( (distancias[3] < distancias[4]) && (distancias[3] < distancias[2]) &&
(distancias[3] < distancias[1]) && (distancias[3] < distancias[0]) )
            menor_distancia_detectada = distancias[3];

        if ( (distancias[2] < distancias[4]) && (distancias[2] < distancias[3]) &&
(distancias[2] < distancias[1]) && (distancias[2] < distancias[0]) )
            menor_distancia_detectada = distancias[2];

        if ( (distancias[1] < distancias[4]) && (distancias[1] < distancias[3]) &&
(distancias[1] < distancias[2]) && (distancias[1] < distancias[0]) )
            menor_distancia_detectada = distancias[1];

        if ( (distancias[0] < distancias[4]) && (distancias[0] < distancias[3]) &&
(distancias[0] < distancias[2]) && (distancias[0] < distancias[1]) )
            menor_distancia_detectada = distancias[0];

//-----botao gravar-----

        if(!digitalRead(botDirGravar))    t_botDirGravar = 0x01;    //Botão Gravar
pressionado? Seta flag

        if(digitalRead(botDirGravar) && t_botDirGravar)                //Botão Gravar solto e
flag setada?
        {
            t_botDirGravar = 0x00;                //Sim...
            lcd.clear();                          //Limpa flag
            lcd.setCursor(4, 0);
            lcd.print("GRAVANDO...");
            delay(100);
            gravarDistancia = menor_distancia_detectada;    //atribui resultado a
variavel para gravar
            Gravar();
        } //end botao gravar

//-----mostrando no LCD sem varredura-----

        lcd.clear();                //Limpa a tela
        lcd.setCursor(0, 0);        //Posiciona o cursor na coluna 0, linha 0;
        lcd.print(distancial,2);    //Envia o valor da variavel distancial para
LCD e duas casas decimais apos a virgula
        lcd.print("m");            //Envia o texto entre aspas para o LCD
        lcd.print(" ");           //Envia um espaco entre aspas para o LCD
        lcd.write((byte)1);       //Mostra o simbolo da flecha cima formado
pelo array
        lcd.setCursor(0, 1);        //Posiciona o cursor na coluna 0, linha 1;
        lcd.print(distancia2,2);    //Envia o valor da variavel distancia2 para
LCD e duas casas decimais apos a virgula
        lcd.print("m");            //Envia o texto entre aspas para o LCD
        lcd.print(" ");           //Envia um espaco entre aspas para o LCD
        lcd.write((byte)2);       //Mostra o simbolo da flecha baixo formado
pelo array
        lcd.setCursor(10, 0);       //Posiciona o cursor na coluna 10, linha 0;
        lcd.print("TOTAL: ");      //Envia o texto entre aspas para o LCD
        lcd.setCursor(10, 1);      //Posiciona o cursor na coluna 10, linha 0;
        lcd.print(menor_distancia_detectada,2); //Envia o valor da variavel distancia_total
para LCD
        lcd.print("m");            //Envia o texto entre aspas para o LCD
    } //end if

//=====Funcao buzzer=====
void Buzzer()
{
    tone(buzzer,1500,1000);        // Ligando o buzzer com uma frequencia de 1500 Hz e
duração de 1000ms.
}

//=====Funcao temperatura com sensor LM35=====
void TemperaturaAmbiente()
{
    temperatura = (float(analogRead(LM35))*5/(1023))/0.01; //Leitura analogica de temperatura do
sensor LM35

    lcd.clear();                //Limpa a tela

```

```

    lcd.setCursor(0, 0); //Posiciona o cursor na coluna 0, linha 0;
    lcd.print("TEMP. AMBIENTE: "); //Envia o texto entre aspas para o LCD
    lcd.setCursor(4, 1); //Posiciona o cursor na coluna 0, linha 1;
    lcd.print(temperatura,2); //Envia o valor da variavel temperatura para
LCD
    lcd.setCursor(9,1); //Posiciona o cursor na coluna 5, linha 1;
    lcd.write((byte)0); //Mostra o simbolo do grau formado pelo
array
    lcd.print("C");
    delay(100);
}

//=====Funcao Data e
hora=====
void Data_hora()
{
    myRTC.updateTime(); //faz leitura do DS1302

    // Imprime informações no LCD

    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("DATA: ");
    lcd.setCursor(6, 0);
    if(myRTC.dayofmonth < 10) lcd.print("0");
    lcd.print(myRTC.dayofmonth);
    lcd.print("/");
    if(myRTC.month < 10) lcd.print("0");
    lcd.print(myRTC.month);
    lcd.print("/");
    lcd.print(myRTC.year);
    lcd.setCursor(0,1);
    lcd.print("HORA: ");
    lcd.setCursor(8,1);
    if(myRTC.hours < 10) lcd.print("0");
    lcd.print(myRTC.hours);
    lcd.print(":");
    if(myRTC.minutes < 10) lcd.print("0");
    lcd.print(myRTC.minutes);
    lcd.print(":");
    if(myRTC.seconds < 10) lcd.print("0");
    lcd.print(myRTC.seconds);
    delay(100);
}

//=====Funcao Consulta Registro=====
void ConsultaRegistro()
{
//-----leitura da ultima gravacao-----

    adress = EEPROM.read(0);
    registro = adress;

    leIndice = EEPROM.get(adress, leIndice);
    adress = adress + 1; //proxima posicao de
memoria

    leituraDistancia = EEPROM.get(adress, leituraDistancia);
    adress = adress + 4;

    leituraTemperatura = EEPROM.get(adress, leituraTemperatura); //temperatura é composta
por 4 bytes
    adress = adress + 4;

    leDiadoMes = EEPROM.get(adress, leDiadoMes);
    adress = adress + 1;

    leMes = EEPROM.get(adress, leMes);
    adress = adress + 1;

    leituraAno = EEPROM.get(adress, leituraAno); //Ano é composta
por 2 bytes, aqui lê o primeiro
    adress = adress + 2;

```

```

leHoras = EEPROM.get(adress, leHoras);
adress = adress + 1;

leMinutos = EEPROM.get(adress, leMinutos);
adress = adress + 1;

leSegundos = EEPROM.get(adress, leSegundos);
adress = adress + 1;

//-----botao Esq-----

if(!digitalRead(botEsq))    t_botEsq = 0x01;           //Botão Esq pressionado? Seta
flag                                                                flag

if(digitalRead(botEsq) && t_botEsq)           //Botão Esq solto e flag
setada?
{ //Sim...
  t_botEsq = 0x00;
  while (flag_esq == 0)
  {
    if(!digitalRead(botEsq))    t_botEsq = 0x01; //Botão Esq pressionado? Seta flag
    if(digitalRead(botEsq) && t_botEsq)           //Botão Esq solto e flag setada?
    {                                             //Sim...
      t_botEsq = 0x00;
      //flag_esq = 1;
      registro -= 16;
      adress = registro;
    }

    if(!digitalRead(botDirGravar))    t_botDirGravar = 0x01;
    if(digitalRead(botDirGravar) && t_botDirGravar)
    {
      t_botDirGravar = 0x00;
      //flag_esq = 1;
      registro += 16;
      adress = registro;
    }

    if(!digitalRead(botUp))    t_botUp = 0x01;
    if(digitalRead(botUp) && t_botUp)
    {
      t_botUp = 0x00;
      flag_esq = 1;
    }

    if(!digitalRead(botDown))    t_botDown = 0x01;
    if(digitalRead(botDown) && t_botDown)
    {
      t_botDown = 0x00;
      flag_esq = 1;
    }

    adress = registro;

    leIndice = EEPROM.get(adress, leIndice);
    adress = adress + 1;           //proxima posicao de
memoria

    leituraDistancia = EEPROM.get(adress, leituraDistancia);
    adress = adress + 4;

    leituraTemperatura = EEPROM.get(adress, leituraTemperatura); //temperatura é
composta por 4 bytes
    adress = adress + 4;

    leDiadoMes = EEPROM.get(adress, leDiadoMes);
    adress = adress + 1;

    leMes = EEPROM.get(adress, leMes);
    adress = adress + 1;

    leituraAno = EEPROM.get(adress, leituraAno);           //Ano é
composta por 2 bytes, aqui lê o primeiro
    adress = adress + 2;

    leHoras = EEPROM.get(adress, leHoras);

```

```

    adress = adress + 1;

    leMinutos = EEPROM.get(adress, leMinutos);
    adress = adress + 1;

    leSegundos = EEPROM.get(adress, leSegundos);
    adress = adress + 1;

    lcd.clear(); //Limpa a tela
    lcd.setCursor(0, 0); //Posiciona o cursor na coluna 0,
linha 0;
    if(leDiadoMes < 10) lcd.print("0");
    lcd.print(leDiadoMes);
    lcd.print("/");
    if(leMes < 10) lcd.print("0");
    lcd.print(leMes);
    lcd.print("/");
    lcd.print(leituraAno);
    lcd.setCursor(11, 0); //Posiciona o cursor na coluna 11,
linha 0;
    lcd.print(leituraDistancia,2);
    lcd.print("m");
    lcd.setCursor(0, 1); //Posiciona o cursor na coluna 0,
linha 1;
    if(leHoras < 10) lcd.print("0");
    lcd.print(leHoras);
    lcd.print(":");
    if(leMinutos < 10) lcd.print("0");
    lcd.print(leMinutos);
    lcd.print(":");
    if(leSegundos < 10) lcd.print("0");
    lcd.print(leSegundos);
    lcd.setCursor(10,1); //Posiciona o cursor na coluna 0,
linha 0;
    lcd.print(leituraTemperatura,1);
    lcd.setCursor(14,1); //Posiciona o cursor na coluna 5,
linha 1;
    lcd.write((byte)0); //Mostra o simbolo do grau formado
pelo array
    lcd.print("C");
    delay(100);

    } //end while
    flag_esq = 0;

} //end if

//-----consulta registro para o LCD-----
    lcd.clear(); //Limpa a tela
    lcd.setCursor(0, 0); //Posiciona o cursor na coluna 0, linha 0;
    if(leDiadoMes < 10) lcd.print("0");
    lcd.print(leDiadoMes);
    lcd.print("/");
    if(leMes < 10) lcd.print("0");
    lcd.print(leMes);
    lcd.print("/");
    lcd.print(leituraAno);
    lcd.setCursor(11, 0); //Posiciona o cursor na coluna 11, linha 0;
    lcd.print(leituraDistancia,2);
    lcd.print("m");
    lcd.setCursor(0, 1); //Posiciona o cursor na coluna 0, linha 1;
    if(leHoras < 10) lcd.print("0");
    lcd.print(leHoras);
    lcd.print(":");
    if(leMinutos < 10) lcd.print("0");
    lcd.print(leMinutos);
    lcd.print(":");
    if(leSegundos < 10) lcd.print("0");
    lcd.print(leSegundos);
    lcd.setCursor(10,1); //Posiciona o cursor na coluna 0, linha 0;
    lcd.print(leituraTemperatura,1);
    lcd.setCursor(14,1); //Posiciona o cursor na coluna 5, linha 1;
    lcd.write((byte)0); //Mostra o simbolo do grau formado pelo
array

```

```

    lcd.print("C");
    delay(100);
}

//=====Funcao Varredura=====
void Varredura()
{
    distancia_varrida = 0;
    while(countVarrer<40)
    {
        TriggersensorME007ULS();           //Dispara o trigger (gatilho para
o sensor ME007                             //Chama a função das leituras
ultrassonicas
        distancia_varrida = distancia_varrida + menor_distancia_detectada;
        countVarrer++;
    }
    countVarrer = 0;

    distancia_varrida_final = distancia_varrida / 40;           //tira media de 40 leituras

    lcd.clear();           //Limpa a tela
    lcd.setCursor(0, 0);   //Posiciona o cursor na coluna 0, linha 0;
    lcd.print(distancia1,2); //Envia o valor da variavel distancial para LCD e
duas casas decimais apos a virgula
    lcd.print("m");       //Envia o texto entre aspas para o LCD
    lcd.print(" ");       //Envia um espaco entre aspas para o LCD
    lcd.write((byte)1);    //Mostra o simbolo da flecha cima formado pelo array
    lcd.setCursor(0, 1);   //Posiciona o cursor na coluna 0, linha 1;
    lcd.print(distancia2,2); //Envia o valor da variavel distancia2 para LCD e
duas casas decimais apos a virgula
    lcd.print("m");       //Envia o texto entre aspas para o LCD
    lcd.print(" ");       //Envia um espaco entre aspas para o LCD
    lcd.write((byte)2);    //Mostra o simbolo da flecha baixo formado pelo array
    lcd.setCursor(10, 0);  //Posiciona o cursor na coluna 10, linha 0;
    lcd.print("TOTAL: "); //Envia o texto entre aspas para o LCD
    lcd.setCursor(10, 1);  //Posiciona o cursor na coluna 10, linha 0;
    lcd.print(distancia_varrida_final,2); //Envia o valor da variavel distancia_varrida_final
para LCD
    lcd.print("m");       //Envia o texto entre aspas para o LCD

    Buzzer();           //chama funcao buzzer
}

//=====Funcao Gravar na EEPROM=====
void Gravar()
{
    myRTC.updateTime();
    int i = 16 ;

    while ( (EEPROM.read(i) != 0xFF) && (i < EEPROM.length()) )
    {
        i++;
    }

    eeAdress = i;           //atribui a posicao vazia a
variavel eeAdress para iniciar sequencia de gravacoes

    valorUltimaGravacao = i;
    EEPROM.put(0, valorUltimaGravacao);
    EEPROM.put(2, 0);
    EEPROM.put(3, 0);
    EEPROM.put(4, 0);
    EEPROM.put(5, 0);
    EEPROM.put(6, 0);
    EEPROM.put(7, 0);
    EEPROM.put(8, 0);
    EEPROM.put(9, 0);
    EEPROM.put(10, 0);
    EEPROM.put(11, 0);
    EEPROM.put(12, 0);
    EEPROM.put(13, 0);
    EEPROM.put(14, 0);
    EEPROM.put(15, 0);
}

```

```

    valorIndice = eeAddress; //Alocar mais um byte para completar
16 bytes * 256 = 4096 bytes da EEPROM do arduino MEGA
    EEPROM.put(eeAddress, valorIndice);
    eeAddress = eeAddress + 1;

    valorDistancia = gravarDistancia; //gravarDistancia ocupa 4 bytes
    EEPROM.put(eeAddress, valorDistancia); //grava a variavel com 4 bytes
iniciando na posicao eeAddress
    eeAddress = eeAddress + 4; //Alocar o eeAddress para depois
de 4 posicoes de memoria

    valorTemperatura = temperatura; //temperatura ocupa 4 bytes
    EEPROM.put(eeAddress, valorTemperatura);
    eeAddress = eeAddress + 4; //Alocar o eeAddress para depois
de 4 posicoes de memoria

    valorDiadoMes = myRTC.dayofmonth;
    EEPROM.put(eeAddress, valorDiadoMes); //Alocar o eeAddress para depois
de 1 posicao de memoria

    valorMes = myRTC.month;
    EEPROM.put(eeAddress, valorMes); //Alocar o eeAddress para depois
de 1 posicao de memoria

    valorAno = myRTC.year; //myRTC ocupa 2 bytes
    EEPROM.put(eeAddress, valorAno);
    eeAddress = eeAddress + 2; //Alocar o eeAddress para depois
de 2 posicoes de memoria

    valorHoras = myRTC.hours;
    EEPROM.put(eeAddress, valorHoras); //Alocar o eeAddress para depois
de 1 posicao de memoria

    valorMinutos = myRTC.minutes;
    EEPROM.put(eeAddress, valorMinutos); //Alocar o eeAddress para
deopis de 1 posicao de memoria

    valorSegundos = myRTC.seconds;
    EEPROM.put(eeAddress, valorSegundos); //Alocar o eeAddress para
deopis de 1 posicao de memoria

    if (eeAddress == EEPROM.length())
    {
        eeAddress = 16;
    }
}

```